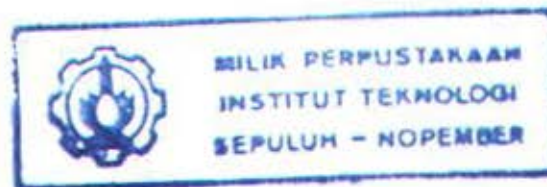


23.271/H/05



TUGAS AKHIR
LS 1336

**ANALISA AKSI KONTROL PADA PENGATURAN PUTARAN MOTOR
DC DENGAN MENGGUNAKAN MODEL MSW 150**

RSSP
629.8
Hak
a-1
2005



Oleh :

M.LUQMAN HAKIM
4299 100 035

PERPUSTAKAAN ITS	
Tgl. Terima	5-4-2005
Terima Dari	H
No. Agenda Prp.	22129

JURUSAN TEKNIK SISTEM PERKAPALAN
FAKULTAS TEKNOLOGI KELAUTAN
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2005

**ANALISA AKSI KONTROL PADA PENGATURAN PUTARAN MOTOR
DC DENGAN MENGGUNAKAN MODEL MSW 150**

TUGAS AKHIR

**Diajukan guna memenuhi syarat untuk
memperoleh gelar Sarjana Teknik
pada**

**Jurusan Teknik Sistem Perkapalan
Fakultas Teknologi Kelautan
Institut Teknologi Sepuluh Nopember Surabaya**

Surabaya, Februari 2005

Menyetujui,

Dosen Pembimbing I



Eddy Setyo K, ST, M.Sc
NIP. 132 133 978



Dosen Pembimbing II



Indra Ranu K, ST, M.Sc
NIP. 132 306 356

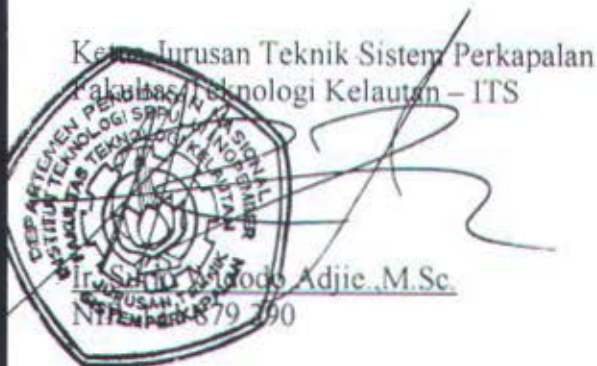


SURAT KEPUTUSAN Pengerjaan Tugas Akhir (LS 1336)

Sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik pada Jurusan Teknik Sistem Perkapalan Fakultas Teknologi Kelautan ITS, maka perlu diterbitkan Surat Keputusan Pengerjaan Tugas Akhir yang memberikan tugas kepada mahasiswa tersebut dibawah ini untuk mengerjakan Tugas Akhir sesuai judul dan lingkup bahasan yang telah ditentukan.

Nama Mahasiswa : M.Luqman Hakim.
NRP : 4299 100 035
Dosen Pembimbing : 1. Eddy Setyo K, ST,M Sc
2. Indra Ranu.K, ST,M Sc
Tanggal diberikan tugas :
Tanggal diselesaikan tugas :
Judul Tugas akhir :

ANALISA AKSI KONTROL PADA PENGATURAN PUTARAN MOTOR DC DENGAN MENGGUNAKAN MODEL MSW 150



Yang menerima tugas :

Mahasiswa

M.Luqman Hakim
NRP. 4299 100 035

Dosen Pembimbing I

Eddy Setyo K, ST, M Sc
NIP. 132 133 978

Dosen Pembimbing II

Indra Ranu K, ST, M Sc
NIP. 132 306 356



LEMBAR KEMAJUAN TUGAS AKHIR

Nama / NRP : M.Luqman Hakim. / 4299 100 035
Judul Tugas Akhir : ANALISA AKSI KONTROL PADA PENGATURAN
PUTARAN MOTOR DC DENGAN MENGGUNAKAN MODEL MSW 150
Dosen Pembimbing : 1. Eddy Setyo K .ST,M.Sc
2. Indra Ranu K,ST,M.Sc

NO	TGL	KEGIATAN	RENCANA ASISTENSI BERIKUTNYA TGL	PARAF DOSEN
1		Asistensi Bab I		
2		Asistensi Bab II & III		
3		Revisi Bab II & III		
7		Asistensi Bab I ; II, III		
5		Revisi Bab I, II, III		
6		Asy. bab IV		
7		Revisi bab IV & V		
8		Asistensi bab IV		
9		Revisi bab IV		
10		Revisi bab IV		
11		Asistensi bab V & VI		
12		Revisi bab V & VI		

Catatan (diisi oleh dosen pembimbing)

1. Tugas Akhir telah : layak / tidak layak (*)

(**) = coret yang tidak perlu

2. Catatan lain yang dianggap perlu: (bila diperlukan bisa menggunakan halaman kosong dibaliknya)

Eddy Setyo K

KATA PENGANTAR

Dengan mengucapkan puji syukur Alhamdulillah kepada **Allah SWT** yang telah memberikan rahmat dan hidayahnya berupa kesehatan dan ilmu pengetahuan sehingga penulis dapat menyelesaikan Tugas Akhir ini pada waktu yang telah ditentukan.

Tugas Akhir ini dikerjakan sebagai persyaratan untuk memperoleh gelar Sarjana Teknik di Jurusan Teknik Sistem Perkapalan, Fakultas Teknologi Kelautan, Institut Teknologi Sepuluh Nopember. Di dalam penyusunan dan pengerjaan laporan ini, penyusun banyak menerima bantuan dan masukan. Untuk itu dalam kesempatan ini penulis mengucapkan terima kasih kepada :

1. Bapak Ir. Suryo Widodo Adjie, M.Sc. selaku Ketua Jurusan Teknik Sistem Perkapalan yang telah memberikan arahan dan tuntunannya dalam penyelesaian Tugas Akhir ini.
2. Bapak Eddy Setyo K,ST, M.Sc. selaku dosen pembimbing Tugas Akhir I, yang telah dengan sabar dan bijaksana memberikan bimbingannya dalam penyelesaian Tugas Akhir ini.
3. Bapak Indra Ranu K, ST. M.Sc selaku dosen pembimbing Tugas Akhir II,, yang telah bersedia meluangkan banyak waktu sampai lembur dalam memberikan bimbingannya dalam penyelesaian Tugas Akhir ini.
4. Bapak Ir.Aguk Zuhdi MF,M.Eng,selaku dosen wali yang telah membimbingku dan memberikan segala pengarahan semenjak masuk di Jurusan Teknik Sistem Perkapalan.

5. Bapak-bapak Dosen beserta Staf Jurusan Teknik Sistem Perkapalan yang tidak dapat disebutkan satu-persatu, yang telah banyak memberikan ilmu pengetahuan, bimbingan dan bantuannya.
6. Ayahanda, Ibunda, yang telah memberikan doa restunya sehingga penulis mendapat kelancaran dalam pengerjaan Tugas Akhir ini.
7. Mbak Elvi, Mas Jemmy dan si kecil Fafa yang telah memberikan dukungan moral dan financial dalam penyelesaian Tugas Akhir ini
8. Mbak Lilik, Mas Andys serta si cantik Nana yang telah banyak membantu baik dari dukungan moral, pikiran dan financial dalam penyelesaian Tugas Akhir ini.
9. Agung, dwi, didin beserta teman-teman kosku yang telah rela meluangkan waktu dan kesabarannya untuk membantu dalam pengerjaan Tugas Akhir ini.
10. Niko Agung yang jadi teman seperjuangan di lab, segenap angkatan 99 yang telah membantu saya dalam berbagai masalah baik secara langsung maupun tidak langsung dalam penyelesaian Tugas Akhir ini.

Dalam penyusunan Tugas Akhir ini penulis menyadari bahwa masih jauh dari yang diharapkan untuk itu kritik dan saran yang bersifat membangun akan selalu saya terima demi sempurnanya laporan ini.

Tidak lupa penulis mohon maaf atas segala kekurangan. Akhirnya penulis hanya bisa berharap semoga laporan ini dapat bermanfaat bagi penulis dan para pembaca. Tidak lupa penulis juga berdoa' a semoga amal ibadah dan kebaikan budi semua pihak yang telah membantu dan membimbing penulis dalam mengerjakan Tugas Akhir ini, senantiasa mendapat ridlo dari Allah.SWT. Amien...

Surabaya, Februari 2005

Penyusun



ABSTRAK

Untuk mengetahui suatu system control menggunakan aksi control yang tepat maka perlu dilakukan suatu analisa dinamik terhadap system control tersebut. Sebelum melakukan analisa terhadap karakteristik dinamik dari suatu system control terlebih dahulu dilakukan pengujian terhadap karakteristik static dari instrument system control tersebut. Dari karakteristik dinamik yang dihasilkan dari setiap aksi control nantinya akan diketahui aksi control yang memberikan hasil yang optimal yaitu aksi control yang memberikan hasil optimal yang dapat memperkecil kesalahan sampai nol atau nilai yang masih dapat ditoleransi, memiliki kecepatan respon cukup cepat dan menunjukkan peredaman yang layak serta memiliki kestabilan mutlak dan kestabilan relative yang layak.

DAFTAR ISI

ABSTRAK

BAB I . PENDAHULUAN

I-1

1.1 Latar Belakang

I-1

1.2 Perumusan Masalah

I-3

1.3 Tujuan Penulisan

I-3

1.4 Batasan Masalah

I-4

1.5 Sistematika Penulisan

I-5

BAB II . DASAR TEORI

II-1

2.1 Sistem Pengendalian

II-1

2.2 Elemen – elemen Sistem Pengendalian Otomatis

II-2

2.3 Sistem Pengukuran

II-7

2.3.1. Elemen Sensing dan Transmitter

II-7

2.3.2. Skala Pengukuran

II-8

2.3.3. Accuracy

II-9

2.3.4. Dead Band

II-10

2.3.5. Saturation dan Dead Zone

II-12

2.4 Pengendali Akhir

II-13

2.5 Aksi Kontrol

II-15

2.5.1. Aksi Kontrol Dua posisi atau Pengendali on- off

II-15

2.5.2. Pengendali Proportional

II-18

2.5.3. Pengendali Integral	II-20
2.5.4. Pengendali Proportional ditambah Integral	II-20
2.5.5. Pengendali Proportional ditambah Integral ditambah Turunan (PID)	II-21
2.6 Kompensator	II-22
2.7 Respon Transien	II-24
2.8 Kriteria Kestabilan	II-28
2.8.1. Kriteria Kestabilan Routh	II-28
2.8.2. Kriteria Kestabilan Root-Locus	II-30
 BAB III. METODOLOGI	 III-1
 BAB IV. BLOK DIAGRAM DAN MODEL MATEMATIK	 IV-1
4.1 Blok Diagram	IV-1
4.2 Model Matematik	IV-1
 BAB V. ANALISA DATA DAN PEMBAHASAN	
5.1 Op-Amp	V-1
5.2 Characteristic Motor DC	V-6
5.3 Error Channel	V-10
5.4 Closed Loop Position Control	V-13
5.5.Simple Speed Control System	V-16
5.6 Dead Band	V-25
5.7 Basic Test	V-27
5.8 Static Characteristic dari Motor DC	V-29

5.9 Frequency Characteristic Dari Motor DC

V-31

5.10 Ziegler & Nichols Method

V- 33

5.11 PID Controler

V-34

LAMPIRAN

BAB I

PENDAHULUAN

I.1 Latar Belakang

Seiring dengan kemajuan teknologi dalam dunia industri yang semakin pesat dan meningkatkan persaingan diantara mereka, pabrik – pabrik membutuhkan peralatan – peralatan otomatis untuk meningkatkan proses produksinya. Untuk mengendalikan parameter proses produksi diperlukan otomatisasi peralatan – peralatannya. Otomatisasi tidak saja diperlukan demi kelancaran operasi, keamanan, ekonomi maupun mutu produk, tetapi lebih merupakan kebutuhan pokok. misalnya kontrol otomatis pada alat – alat mesin di dunia manufaktur, kontrol tekanan dalam vessel atau pipa, aliran dalam pipa, suhu dalam heat exchanger, kontrol kecepatan motor, dll. Karena kemajuan dalam teori dan praktek kontrol otomatis maka dapat mempertinggi kualitas dan menurunkan biaya produksi, mempertinggi laju produksi, meniadakan pekerjaan – pekerjaan rutin dan membosankan yang harus dilakukan oleh manusia.

Servomekanisme banyak digunakan dalam dunia industri modern, sebagai contoh operasi mesin perkakas yang seluruhnya otomatis, bersama-sama dengan instruksi yang diprogram terlebih dahulu, dapat dibuat dengan menggunakan servomekanisme, seperti pada pengendalian terhadap putaran motor yang akan dibahas dalam tugas akhir ini. Pada pengendalian kecepatan putaran motor oleh manusia atau system pengendalian manual, masih tetap dipakai pada aplikasi tertentu. Biasanya untuk system yang tidak banyak mengalami perubahan load

(beban) atau pada proses yang tidak kritis. Load didalam pengendalian diatas adalah kecepatan putaran dari motor tidak diubah-ubah oleh operator. Kalau proses kerja dari motor tidak perlu berubah-ubah kecepatan putarannya maka operator tidak perlu setiap waktu untuk mengamati dan melakukan tindakan koreksi terhadap perubahan kecepatan. Kesalahan/keteledoran dari operator akan menyebabkan waktu kerja semakin lama karena putaran motor terlalu rendah dan kerusakan motor jika putaran motor terlalu tinggi. Operator adalah manusia sehingga sangat mudah dipengaruhi oleh menurunnya konsentrasi kerja, rasa jenuh dan keadaan kritis lainnya.

Untuk itu system harus dibuat secara otomatis. Peran dari operator didalam system pengendalian manual digantikan oleh sebuah alat yang disebut controller. Tugas untuk mengatur kecepatan motor tidak lagi dilakukan oleh manusia/operator tapi oleh atas perintah controller.

Sistem Kontrol terdiri atas kontroler otomatis, plant dan sensor (elemen pengukur). Kontroler mendeteksi sinyal kesalahan aktuatisasi, yang biasanya mempunyai tingkat daya juang sangat rendah dan memperkuatnya menjadi tingkat yang tingginya mencukupi jadi kontroler otomatis terdiri dari detector kesalahan dan penguat atau amplifier actuator yang merupakan alat daya yang menghasilkan masukan ke plant sesuai dengan sinyal control sedemikian sehingga sinyal umpan balik akan berkaitan dengan sinyal masukan acuan. Keluaran dari kontroler otomatis dimasukkan ke dalam actuator. Sensor atau elemen pengukur adalah alat yang mengubah variable yang sesuai seperti kecepatan, tegangan yang dapat digunakan membandingkan keluaran dengan sinyal masukan (setting point).

1.2 Perumusan Masalah

Dalam tugas akhir ini permasalahan yang akan dianalisa adalah :

1. Pemilihan aksi kontrol untuk mengatur putaran motor DC didasarkan time response dari system tersebut, pada umumnya untuk melakukan pemilihan dilakukan secara teoritis, oleh karena itu dalam Tugas Akhir ini akan melakukan percobaan aksi kontrol dari P, PI, PID pada alat MSW 150, percobaan ini nantinya diharapkan dapat menjadi acuan dalam pemilihan aksi kontrol yang sesuai
2. Untuk memilih aksi kontrol yang memberikan hasil paling optimal untuk mengontrol kecepatan putaran motor dc yang mampu mengatasi perubahan parameter agar efisiensi meningkat sehingga dapat digunakan sebagai acuan merancang alat kontrol yang berhubungan dengan kecepatan.

1.3 Tujuan Penulisan

Tugas akhir ini bertujuan :

1. Mengetahui karakteristik komponen yang digunakan dari MSW 150.
2. Mengetahui karakteristik dinamik dari tiap – tiap aksi control dipakai P (Proporsional), PI (Proposional + Integral), PID (Proposional + Integral + Derivative) yang digunakan pada alat kontrol putaran MSW 150

1.4 Batasan Masalah

Adapun hal yang menjadi batasan dalam penulisan ini adalah :

- a. Sinyal masukan yang dikenakan pada analisa respon transien dan kesalahan tunak adalah sinyal masukan step response.
- b. Pengaruh gangguan pada sistem kontrol dari luar (lingkungan) diabaikan.
- c. Analisa kesalahan yang digunakan adalah steady state error dari sistem.
- d. Peralatan & komponen dari MSW 150 diasumsikan dalam keadaan baik.

1.5 Sistematika Penulisan

Bab I Pendahuluan

Pada bab ini akan dibahas mengenai latar belakang penulisan tugas akhir, perumusan masalah, pembatasan masalah, tujuan tugas akhir dan sistematika penulisan.

Bab II Dasar Teori

Yang berisi prinsip – prinsip sistem pengendalian yang juga meliputi tentang Kontrol Proportional (P), Kontrol Proportional plus Integral (PI), Kontrol Proportional plus Integral plus Derivativ (PID).

Bab III Metodologi

Yang berisi metode dan langkah dalam pengerjaan tugas akhir

Bab IV Pengembangan model matematis dan blok diagram dari MSW

150

Yang berisi pengembangan model matematis dalam bentuk m.files dan blok diagram.

Bab V Analisa Data dan Pembahasan

Yang berisi tentang penyimulasian dan analisa hasil perhitungan berbasis komputer dengan penerapan aksi Kontrol Proportional(P), Kontrol Proportional plus Integral (PI), Kontrol Proportional plus Integral plus Derivativ (PID). Yang digunakan untuk mengontrol kecepatan dengan menggunakan alat kontrol MSW 150.

Bab VI Penutup

Pada bab ini akan berisi kesimpulan dari tugas akhir yang telah selesai dikerjakan dan saran mengenai permasalahan yang ada.



BAB II

DASAR TEORI

II.1. Sistem Pengendalian

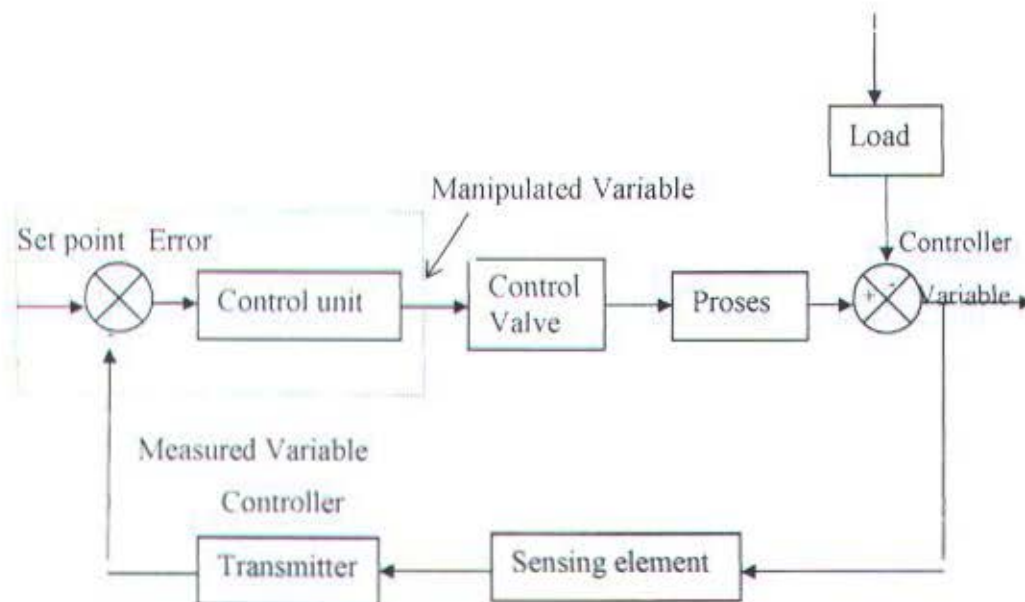
Sistem Pengendalian merupakan gabungan serta kerja alat-alat pengendalian otomatis. Dalam pengendalian proses banyak parameter yang dikendalikan, yang umum biasanya yaitu pressure (tekanan), aliran (flow), suhu (temperature), level, kecepatan dan masih banyak yang lain. Pada umumnya saat ini pengendalian dilakukan oleh manusia sebagai operator, contoh : pada pengendalian level (permukaan zat cair) di sebuah tangki, operator disini fungsinya sebagai kontroler dimana dia melakukan tugas untuk mengamati level dari tangki dan melakukan tindakan mengurangi atau menambahkan jika level dalam tangki tidak sesuai set point (yang dikehendaki). Pengendalian proses operator mengerjakan empat langkah berikut :

Mengukur – Membandingkan – Menghitung- Mengoreksi.

Seorang operator yang berpengalaman tidak akan sembarang melakukan tindakan dalam kerjanya pada control level diatas, ia akan memperkirakan seberapa banyak valve yang perlu dibuka atau ditutup. Pada tahapan ini sebenarnya operator sedang melakukan tahapan **mengukur** dimana dia mengamati ketinggian dari level dari tangki. kemudian operator **membandingkan** hasil pengukurannya apakah sudah sesuai dengan keinginan. kemudian operator **menghitung** berapa banyak jumlah valve yang perlu dibuka atau ditutup. Langkah berikutnya adalah mengubah bukaan valve sesuai dengan hasil perbandingan dan perhitungan tadi. Langkah tersebut disebut dengan langkah **mengkoreksi**.

II.2. Elemen-elemen Sistem Pengendalian Otomatis

Didalam diagram kotak sistem pengendalian otomatis akan selalu ada komponen-komponen pokok seperti elemen proses, elemen pengukuran (sensing element dan transmitter), elemen pengukur (sensing element dan transmitter), elemen kontroler (control unit) dan final control element (control valve) dalam bentuk matematis, semua kotak element itu kelak akan diisi persamaan-persamaan matematik yang merupakan transfer function elemen-elemen tersebut.



Gambar 1.1. Block Diagram system pengendalian otomatis

KETERANGAN :

- *Set point* adalah besar process variable yang dikehendaki. Sebuah controller akan selalu berusaha untuk menyamakan controller variable dengan set point.

- *Measurement variable* (measured variable) adalah sinyal yang keluar dari transmitter. Besaran ini merupakan cerminan besarnya sinyal sistem pengukuran.
- *Error* adalah selisih antara *Set Point* dikurangi *Measurement variable*. Error bisa negatif dan juga bisa positif, sebaliknya bila set point lebih kecil dari measured variable, error akan menjadi positif.
- *Control unit* adalah bagian dari controller yang menghitung besarnya koreksi yang diperlukan. Input control unit adalah error dan outputnya adalah sinyal yang keluar dari controller (*manipulated variable*). Control unit memiliki transfer function yang tergantung pada jenis controller. Output control unit adalah hasil penyelesaian matematik transfer function dengan memasukkan nilai error sebagai input.
- *Controller* adalah elemen yang mengerjakan tiga dari empat tahap langkah pengendalian, yaitu membandingkan set point dengan measurement variable, menghitung berapa banyak koreksi yang perlu dilakukan dan mengeluarkan sinyal koreksi sesuai dengan hasil perhitungan tadi. Controller sepenuhnya menggantikan peran manusia dalam mengendalikan sebuah proses.
- *Manipulated variable* adalah input dari suatu proses yang dapat dimanipulasi atau diubah-ubah besarnya agar process variable atau controlled variable besarnya sama dengan set point.
- *Final control variable* (*Control valve*) adalah bagian akhir dari instrumentasi system pengendalian. Bagian ini berfungsi untuk

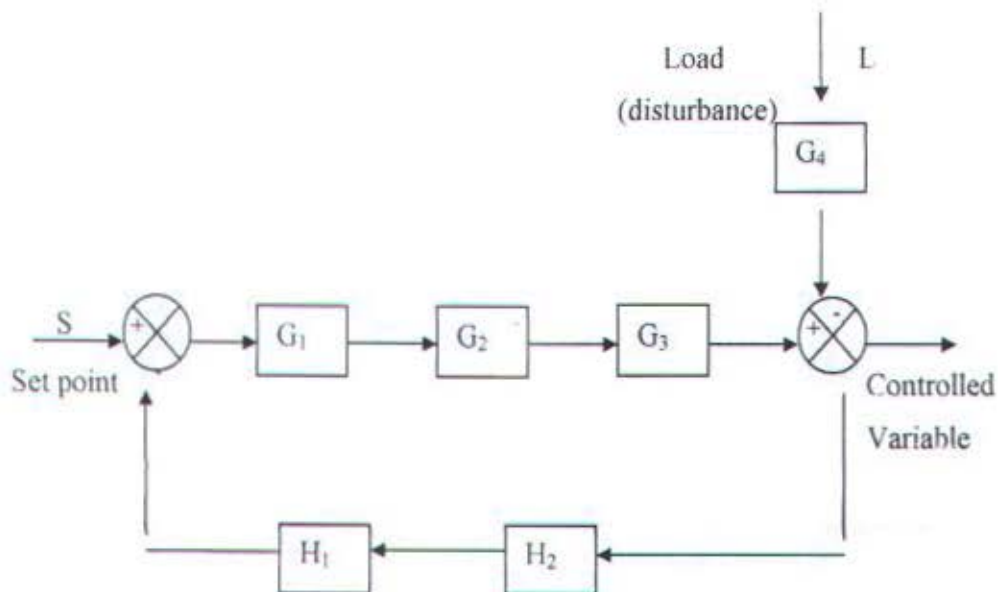
mengubah measurement variable dengan cara memanipulasi besarnya manipulated variable, berdasarkan perintah controller.

- *Proses* adalah tatanan peralatan yang mempunyai suatu fungsi tertentu. Input proses dapat bermacam-macam yang pasti ia merupakan besaran yang dimanipulasi oleh final control element atau control valve agar measurement variable sama dengan set point.
- *Disturbance (Load)* adalah besaran lain, selain manipulated variable yang dapat menyebabkan berubahnya controlled variable.
- *Controlled variable* adalah besaran atau variable yang dikendalikan. Besaran ini pada diagram kotak juga disebut output proses atau proses variable.
- *Sensing element (Sensor)* adalah bagian paling ujung suatu system pengukuran (measuring system).
- *Transmitter* adalah alat yang berfungsi untuk membaca sinyal sensing element, dan mengubahnya menjadi sinyal yang dapat dimengerti oleh controller.

Didalam Gambar 1.1 bagian controller mempunyai *summing junction* dengan tanda positif-negatif (+ / -). Di titik ini langkah membandingkan dilakukan dengan mengurangi besaran set point dengan sinyal measurement variable. Hasilnya adalah sinyal yang disebut error. Tanda negatif (-) di *summing junction* membawa arti yang sangat spesifik bagi seluruh system. Karena tanda inilah system pengendalian otomatis juga lazim disebut system negative feedback. Dengan demikian, system pengendalian otomatis mempunyai dua nama lain, yaitu

system closed loop dan system negative feedback. Pada system feedback positif, tanda negatif (-) di *summing junction* diganti dengan tanda positif (+). Jadi ada dua tanda positif (+) di *summing junction*. Set point tidak lagi dikurangi sinyal *measurement variable*. Jadi, error bukan set point dikurangi *measurement variable*, melainkan set point ditambah sinyal *measurement variable*.

Secara matematis, diagram kotak system closed loop yang ada pada Gambar 1.1 dapat digambar ulang menjadi bentuk yang ada didalam Gambar 1.2. Kedua diagram kotak tersebut sebenarnya sama, hanya saja yang satu tampil dalam bentuk simbolis dan yang lain tampil dalam bentuk matematis. Isi masing-masing kotak untuk sementara diisi transfer function bentuk umum (G_1 , G_2 , G_3 , G_4 , H_1 dan H_2) yang masing-masing mewakili : controller, control valve, proses, pengaruh load, transmitter dan sensing element.



Gambar 1.2. Block diagram umum system pengendalian

G_1 = Control unit

G_2 = Final control element

G_3 = Proses

G_4 = Disturbance (Load)

H_1 = Transmitter

H_2 = sensing element

Dari Block diagram diatas, diturunkan persamaan yang menyatakan hubungan perubahan process variable (C) yang terjadi karena adanya perubahan set point (S) atau perubahan Load (L). Dari jalur proses akan diperoleh sinyal $C = G_1 \cdot G_2 \cdot G_3 \times S$; dan dari jalur Load akan diperoleh sinyal $C = G_4 \times L$; dan dari jalur feedback akan diperoleh sinyal $C = C \times G_1 \cdot G_2 \cdot G_3 \cdot H_1 \cdot H_2$. Penggabungan sinyal itu, akan menghasilkan persamaan :

$$C = S (G_1 \cdot G_2 \cdot G_3) - (G_1 \cdot G_2 \cdot G_3 \cdot H_1 \cdot H_2) C - G_4 \times L \quad (3.1)$$

$$C (1 + G_1 \cdot G_2 \cdot G_3 \cdot H_1 \cdot H_2) = S (G_1 \cdot G_2 \cdot G_3) - G_4 \times L \quad (3.2)$$

$$C = S \frac{(G_1 \cdot G_2 \cdot G_3)}{(1 + G_1 \cdot G_2 \cdot G_3 \cdot H_1 \cdot H_2)} - L \frac{G_4}{(1 + G_1 \cdot G_2 \cdot G_3 \cdot H_1 \cdot H_2)} \quad (3.3)$$

Ada dua komponen didalam persamaan 3 diatas, komponen pertama adalah set point (S) disebelah kiri, dan komponen kedua adalah komponen Load (L) disebelah kanan. Dari persamaan ini jelas bahwa Process variable (C) tergantung pada set point (S) dan Load (L). System dikatakan stabil kalau perubahan set point akan menyebabkan response yang overdamped atau underdamped. System dikatakan tidak stabil kalau perubahan set point akan

menyebabkan response yang sustain oscillation atau yang undamped. Hal yang sama juga terjadi atas perubahan Load. System dikatakan stabil kalau karena perubahan load akan menyebabkan response yang overdamped atau underdamped. System dikatakan tidak stabil kalau perubahan Load akan menyebabkan response yang sustain oscillation atau yang undamped.

II.3. Sistem Pengukuran

II.3.1. Elemen Sensing dan Transmitter

Ada dua macam output sinyal system pengukuran, yaitu sinyal elektrik (mA atau V) dan sinyal Pneumatik. Adalagi output sinyal pengukuran dalam bentuk sinyal hidrolik. Pemilihan sinyal pengukuran sangat ditentukan oleh jenis controller yang akan dipakai. Kalau controller yang akan dipakai dari jenis elektronik, tentu akan dipilih sinyal elektrik. Kalau dipilih controller pneumatic, tentu sinyal yang dipilih juga sinyal pneumatic. Untuk menterjemahkan sinyal system pengukuran dari sensing element menjadi sinyal yang dapat dimengerti oleh controller, dibutuhkan sebuah unit yang disebut transmitter. Sebagai standarisasi, sinyal yang keluar dari transmitter, baik elektrik atau pneumatic, dibuat hanya bekerja pada standart skala tertentu. Untuk sinyal pneumatic skala kerjanya 3-15 psig atau $0,2-1 \text{ kg/cm}^2$, dan untuk sinyal elektrik, skala kerjanya ada dalam dua bentuk, yaitu sinyal arus atau sinyal tegangan. Skala kerja sinyal arus selalu 4-20 mA atau sinyal 10-50 mA, dan skala kerja sinyal tegangan agak sedikit bervariasi ada yang 1-5 V_{DC}, ada yang 0-10 V_{DC}, atau skala-skala lain yang tergantung pada kerja unit instrumentasi. Pada umumnya sinyal yang keluar dari transmitter elektronik hampir selalu dalam bentuk sinyal 4-20 mA.

II.3.2. Skala Pengukuran

Setiap system pengukuran mempunyai batasan-batasan skala yang spesifik untuk daerah dimana system beroperasi. Demikian juga dengan satuan pembacaan yang tergantung pada jenis process variable yang diukur, misalnya derajat celcius, psi, kg/cm^2 , gallon per menit dan sebagainya. Skala baca tidak harus dimulai dari nol, dan lebar skala baca tidak perlu terlalu berlebihan karena hanya akan mengorbankan ketelitian pengukuran. Contohnya thermometer suhu badan manusia skala bacanya sudah cukup kalau dimulai dari 35°C sampai 42°C , karena suhu badan manusia memang tidak mungkin berada diluar skala itu. Lagi pula dengan skala baca yang sempit itu, ketelitian sampai $0,1^{\circ}\text{C}$ akan dapat dengan mudah didapat.

Dalam ilmu instrumentasi, ada beberapa istilah yang lazim dipakai untuk menyatakan skala baca. Salah satu diantaranya adalah *zero*. Istilah *zero* dipakai untuk menyatakan titik baca terendah di suatu skala baca. Contohnya, thermometer suhu badan diatas memiliki titik *zero* di 35°C . Istilah yang lain adalah *span*. Istilah ini dipakai untuk menyatakan jarak antara titik baca minimum dan titik baca maksimum. Pada contoh thermometer diatas, *span* adalah 42°C dikurangi 35°C atau sama dengan 7°C . Selain istilah *span* dikenal juga istilah lain yang pengertiannya mirip dengan pengertian *span*, yaitu *range*. *Range* dipakai untuk menyatakan batas-batas daerah kerja. Pada contoh termometer suhu badan diatas, *range* termometer ini dikatakan $35^{\circ}\text{C} - 42^{\circ}\text{C}$.

Didalam praktek, pembacaan semua sinyal system pengendalian selalu dinyatakan dalam bentuk persentasi skala baca. Hal ini dibuat untuk

mempermudah pembacaan pada waktu kalibrasi serta mempermudah pembahasan kerja sinyal-sinyal didalam system pengendalian. Ketidak cocokan atau kekeliruan hubungan satu sinyal dengan sinyal yang lain disebut dengan error. Akan tetapi error disini tidak sama dengan error yang berarti set point dikurangi measurement variable. Error disini berarti sebagai kesalahan baca. Ada banyak amcam error yang terjadi pada unit-unit instrumentasi antara lain *accuracy* (ketelitian), *linierity* (liniaritas), *hysteresis*, *dead band*, *dead zone*, *saturation* dan *repeatability*. Error atau kesalahan-kesalahan baca diatas kelihatannya sangat sepele namun sebenarnya mempunyai potensi yang cukup luas untuk mengacaukan kerja system pengendalian.

II.3.3. Accuracy

Accuracy (akurasi, ketelitian, keakuratan) adalah ketepatan suatu alat ukur dalam memberikan hasil bacaan. Besaran ini menunjukkan banyaknya penyimpangan yang terjadi pada sebuah alat ukur, atau system pengukuran.

Ada beberapa cara untuk menyatakan *Accuracy* :

1. Dalam variabel pengukurannya.

Misalnya, sebuah termometer dengan skala $0^{\circ}\text{F} - 100^{\circ}\text{F}$ disebutkan mempunyai *Accuracy* 1°F . Bila termometer tersebut menunjukkan temperatur 60°F , temperatur yang sebenarnya berkisar antara $59^{\circ}\text{F} - 61^{\circ}\text{F}$.

2. Dalam persentasi span.

Sebuah pressure transmitter memiliki range 100 - 400 psi dan accuracy 0,5 % span. Artinya, setiap sinyal yang keluar dari transmitter tersebut dapat menyimpang sampai $0,50\% \times 300\text{ psi} = 1,5\text{ psi}$. Jadi, kalau hasil pembacaan

transmitter adalah 200 psi, tekanan atau pressure yang sebenarnya berkisar antara 198,5 sampai 201,5 psi.

3. Dalam persentasi terhadap skala maksimum.

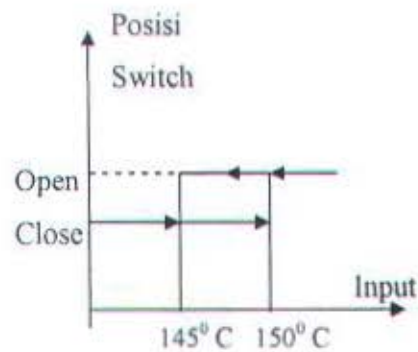
Istilah skala maksimum merupakan terjemahan bebas dari istilah full scale (FS). Sebuah voltmeter dikatakan mempunyai ketelitian 1 % FS, artinya kalau volimeter diletakkan pada skala baca maksimum 200 Volt, ketelitian di range itu adalah ± 2 Volt. Kalau diletakkan di skala baca 1000 Volt, ketelitian di range itu adalah ± 10 Volt.

4. Dalam persentasi pembacaan.

Ketelitian dalam hal ini tergantung pada hasil pembacaan. Sebuah level transmitter dikatakan memiliki *Accuracy* 0,5 % output. Range transmitter tersebut, 0-100". Pada waktu transmitter menunjukkan sinyal 40 % atau 40", level yang sebenarnya berkisar antara 39,8" sampai 40,2". Pada waktu transmitter menunjukkan 60 % atau 60", level yang sebenarnya berkisar antara 57" sampai 63".

II.3.4. Dead Band

Dead band adalah gejala khas yang selalu terjadi pada semua jenis proses switch. Proses switch adalah sakelar yang akan berpindah posisi apabila settingnya tercapai. Contoh proses switch yang sudah dikenal secara umum adalah temperature switch yang ada pada setrika otomatis. Temperature switch itu akan bekerja secara on-off untuk mengendalikan temperatur setrika. Seperti pada Gambar dibawah ini :

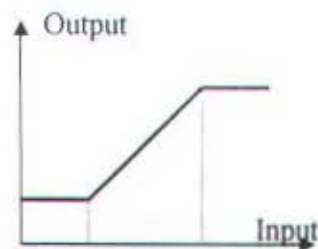


Gambar 1.3. Dead band

Andaikan switch disetel pada setting 150°C . Pada waktu temperatur naik dari 100°C sampai ke 149°C , switch masih belum berpindah posisi. Tetapi begitu temperatur mencapai 150°C , switch segera berpindah dari posisi dari close ke open. Hal yang sama terjadi pada waktu temperatur turun. Posisi switch akan kembali berpindah dari open ke close, hanya perpindahan itu terjadi setelah temperatur ada dibawah 150°C . Gejala inilah yang dimaksud dengan gejala *dead band*. Andaikan perpindahan itu terjadi di 145°C , jarak antara 150°C dan 145°C itu disebut *dead band* atau *black lash*. Jadi *dead band* dapat dipakai untuk menggambarkan suatu keadaan dan dapat juga dipakai untuk menyatakan jarak setting. Semua process switch, misalnya pressure switch, level switch, flow switch pasti mengalami gejala *dead band*. Gejala *dead band* tidak selalu merugikan, bahkan gejala ini sering dimanfaatkan untuk pengendalian on-off. Untuk keperluan pengendalian on-off, lebar sempitnya *dead band* justru dapat dimanfaatkan. Proses switch kemudian dibuat dengan *adjustable dead band*, jarak antara titik high dan titik low bisa disetel. Gejala *dead band* kadang-kadang muncul secara bersamaan dengan gejala hysteresis.

II.3.5. Saturation dan Dead Zone

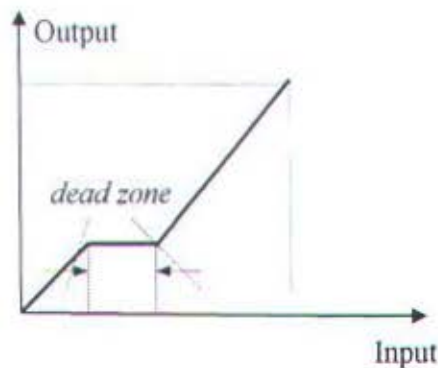
Didalam system pengendalian, sebuah elemen dikatakan saturate bilamana output tidak dapat bertambah lagi bilamana output tidak dapat bertambah lagi walaupun padanya diberikan tambahan input. Gejala saturation dapat terjadi pada semua elemen yang ada didalam system pengendalian. Contoh nyata keadaan saturation (kejenuhan) adalah keadaan yang sering terjadi pada elemen control valve. Misalkan sebuah control valve mempunyai range kerja 3-15 psi untuk bukaan 0-100 %. Kalau pada control valve itu dikenakan sinyal 15 psi, valve akan terbuka 100 %. Pada keadaan itu, control valve tersebut dikatakan *saturate*. Hal yang sama terjadi kalau control valve diberi input dibawah 3 psi. Control valve tidak mungkin lebih tertutup dari 0 %. Jadi keadaan saturate bisa terjadi diatas skala 100 % bisa pula terjadi dibawah skala 0 %. Mekanisme control valve menyebabkan kedua keadaan *saturate* itu. Gambar saturation dapat dilihat seperti dibawah ini :



Gambar 1.4 Saturation

Ada istilah lain yang sering dirancukan dengan istilah *dead band* yaitu *dead zone*. Pengertian *dead zone* memang mirip dengan pengertian *dead band*, khususnya terjadi pada proses switch. Dalam bentuk kurva, gejala *dead zone*

justeru lebih mendekati gejala saturation, hanya saja gejala keadaan saturate adanya ditengah-tengah daerah kerja. Gambar dead zone adalah sebagai berikut :



Gambar 1.5. Dead zone

Pada gambar terlihat didaerah *dead zone*, tambahan input sampai ketitik-titik tertentu tidak akan mengubah output. Proses seolah-olah tidak bekerja atau mati di daerah itu. Elemen yang memiliki *dead zone* akan memiliki Gain yang rendah di daerah *dead zone* itu.

II.4. Pengendali Akhir

Elemen pengendali akhir merupakan terjemahan bebas dari final control element. Elemen ini adalah bagian akhir system pengendalian yang bertugas melakukan langkah koreksi. Ada banyak macam final control element, namun hanya control valve yang paling umum dipakai di system pengendalian. Oleh karena alasan itulah, ada banyak macam pengendali akhir (final control element). Secara umum mereka terbagi menjadi dua bagian, yaitu final control element

untuk pengendalian on-off, misalnya solenoid valve, heating element, electrical contactor atau peralatan elektro mekanik lainnya dan final control element untuk pengendalian kontinu, misalnya control valve dan sebagainya. Control valve adalah jenis final control element yang paling umum dipakai untuk system penegendalian proses, sehingga orang cenderung mengartikan final control element sebagai control valve.

Walaupun ada beberapa system pengendalain on-off yang menggunakan control valve sebagai final control element, namun fungsi control valve di pengendalian itu tidak lebih istimewa dari kerja sebuah solenoid valve. Control valve hanya akan bekerja di dua posisi, yaitu terbuka penuh atau tertutup penuh. Pada pengendalian continous (pengendalian dengan pengendali P, PI, PD atau PID) control valve justru tidak diharapkan berada di posisi tertutup penuh atau terbuka penuh. Control valve harus secara kontinu mengendalikan manipulated variable agar proses variable selalu tetap sama dengan set point. Sebuah control valve terdiri atas dua bagian dasar, yaitu actuator dan valve. Bagian actuator adalah bagian yang mengerjakan gerak buka-tutup valve, dan bagian valve adalah komponen mekanis yang menentukan besarnya flow yang masuk ke proses. Dalam kesatuannya sebagai unit control valve, actuator dan valve harus melakukan tugas koreksi berdasarkan sinyal manipulated variable yang keluar dari controller.

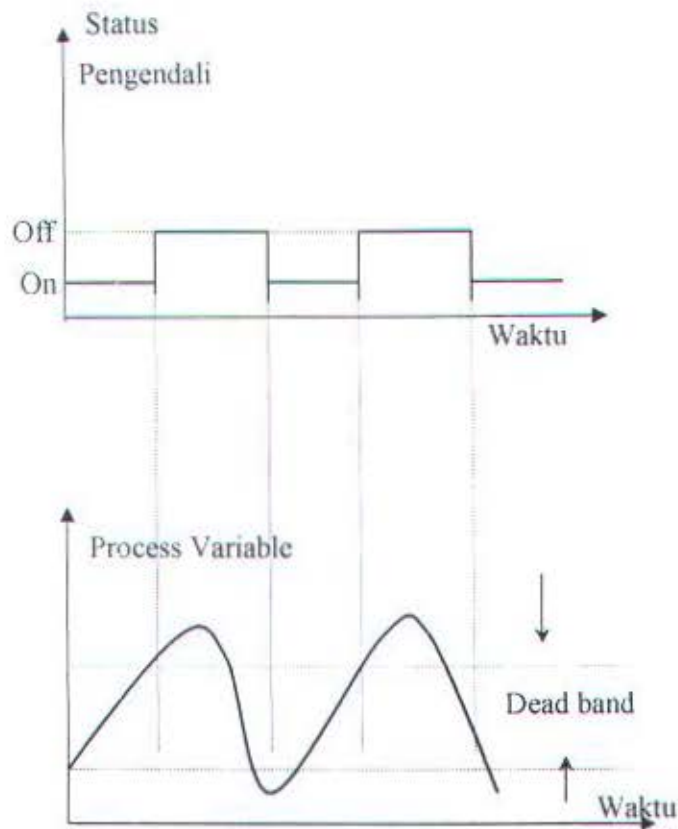
11.5. Aksi Kontrol

Kontroller analog industri dapat diklasifikasikan sesuai dengan aksi pengontrolannya sebagai berikut :

1. Kontroller dua posisi atau on-off.
2. Kontroller Proporsional (P controller).
3. Kontroller Integral (I controller).
4. Kontroller Proporsional plus Integral (PI controller).
5. Kontroller Proporsional plus Derivatif (Pdcontroller).
6. Kontroller Proporsional plus Integral plus Derivatif (PID controller).

11.5.1. Aksi Kontrol Dua Posisi atau Pengendali on-off

Pengendali *on-off* hanya bekerja pada dua posisi, yaitu posisi “on” dan posisi “off”. Kalau final control element berupa control valve, kerja valve hanya terbuka penuh atau tertutup penuh. Pada system pengendalian *on-off*, control valve tidak akan bekerja di daerah antara 0 sampai 100 %. Karena kerjanya yang *on-off*. Hasil pengendalian pengendali *on-off* akan menyebabkan process variable yang bergelombang (tidak pernah konstan). Seperti tampak pada Gambar 3.7 dibawah ini :



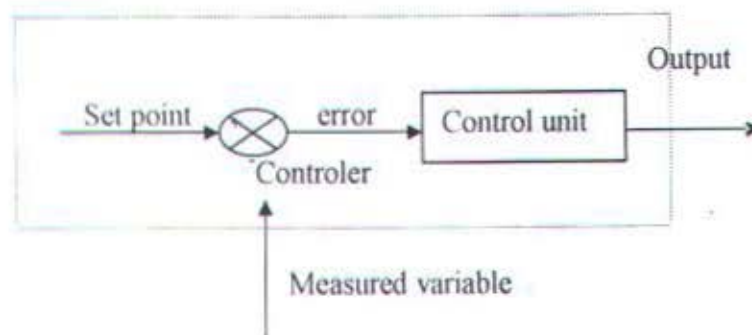
Gambar 1.6. Kerja Controller on-off

Karena karakteristik kerjanya yang hanya on dan off, controller jenis on-off juga sering disebut sebagai two-position controller, gap controller atau snap controller. Kerja pengendalian on-off seringkali didapat dengan memanfaatkan dead band suatu process switch. Kerja pengendali on-off banyak dipakai di system pengendalian yang sederhana karena harganya yang relatif murah. Namun tidak semua proses dapat dikendalikan secara on-off karena banyak operasi proses yang tidak dapat mentolerir fluktuasi process variable. Jadi syarat utama untuk memakai pengendali on-off bukan untuk menghemat biaya pembelian unit

controller, melainkan karena proses memang tidak dapat mentolerir fluktuasi process variable pada batas-batas kerja pengendalian on-off.

Selain pengendali on-off yang bekerja hanya pada dua posisi on dan off, ada pengendali lain yang bekerja secara kontinu. Secara kontinu artinya control valve bekerja pada titik 0 % sampai 100 %. Jenis pengendali ini lebih banyak dipakai di system pengendalian proses, karena banyak operasi proses yang tidak dapat mentolerir fluktuasi process variable yang besar. Ada tiga jenis pengendali kontinu, yaitu pengendali proporsional (P), pengendali Integral (I) dan pengendali diferensial (D).

Karena kelebihan dan kekurangan ketiga pengendali itu, mereka seringkali dipakai dalam bentuk kombinasi, yaitu P + I disingkat PI dan P + I + D disingkat PID. Pada dasarnya, tugas sebuah pengendali kontinu terbagi dalam dua tahap, yaitu membandingkan dan menghitung. Kedua tugas tersebut didalam diagram kotak diwakili oleh kerja summing junction dan kerja unit control seperti terlihat pada Gambar 1.7 :



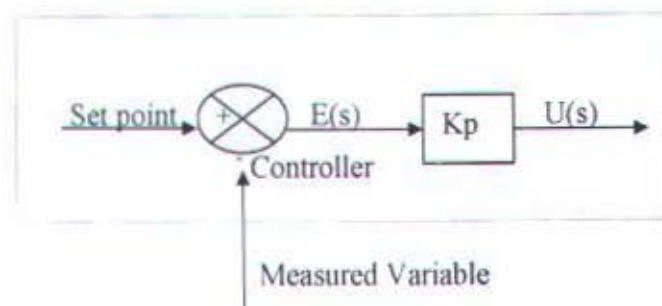
Gambar 1.7. Control unit

Summing junction bertugas membandingkan measured variable dengan set point, dan unit control bertugas menghitung besar kecilnya koreksi yang

diperlukan. Pembandingan itu sendiri dilakukan dengan mengurangi besaran set point dengan besaran measurement variable, yang hasilnya adalah besaran yang disebut error. Karena set point bisa lebih kecil atau lebih besar dari measurement variable, nilai error bisa negatif atau positif. Berdasarkan besarnya error inilah unit control menghitung besarnya koreksi. Error adalah input unit kontrol dan manipulated variable adalah output unit control. Besarnya manipulated variable dihitung berdasarkan besarnya error dan transfer function unit control. Bentuk transfer function dari unit control tergantung pada "mode" yang ada di controller (control mode) : P, I, PI, PD dan PID.

II.5.2. Pengendali Proporsional

Salah satu dari ketiga mode unit control yang paling populer dan paling banyak dipakai adalah unit control P. Seperti tercermin dari namanya, besar output unit control P selalu sebanding dengan besarnya input. Unit control P adalah unit control yang paling banyak dipakai, baik tersendiri dalam bentuk pengendali P-only maupun dalam bentuk kombinasi dengan mode integral (I) dan diferensial (D). Bentuk transfer function aksi kontrol proporsional adalah sebagai berikut :



Gambar 1.8. Pengendali proporsional.

Untuk kontroller dengan aksi kontrol proporsional, hubungan antara masukan kontroller $u(t)$ dan sinyal pembangkit kesalahan $e(t)$ adalah sebagai berikut (Ogata, 1997) :

$$u(t) = K_p e(t)$$

Atau dalam transformasi Laplace

$$\frac{U(s)}{E(s)} = K_p \quad (3.4)$$

dengan K_p adalah suku penguatan proporsional atau Gain proporsional.

Gain (K_p) unit control proporsional bisa berupa bilangan bulat, bilangan pecahan, positif atau juga berharga negatif. Yang pasti besarnya tetap linear di semua daerah kerja dan tidak tergantung pada fungsi waktu. Sepintas istilah Gain memberikan kesan bahwa ada penguatan atau pembesaran sinyal. Padahal gain bisa saja berbentuk bilangan pecahan, bahkan negatif. Sehingga output bisa lebih kecil dari input, dan juga bisa menjadi negatif. Oleh karena alasan itu, didalam praktek istilah gain jarang dipakai, dan yang lazim adalah istilah *Proportional band* (PB), dimana menurut Gunterus didapatkan persamaan :

$$K_p = \frac{100\%}{PB} \quad (3.5)$$

Jadi, kalau PB sama dengan 50 %, gain sama dengan 2; jika PB sama dengan 200 %, gain sama dengan $\frac{1}{2}$; jika gain sama dengan 100 %, gain tepat sama dengan 1. Karena PB berbanding terbalik terhadap gain, kalau PB semakin besar gain akan semakin kecil. Kedua besaran itu begitu pentingnya karena mereka memberikan pengaruh langsung pada kestabilan system.

II.5.3. Pengendali Integral

Pada kontroller dengan aksi integral nilai masukan kontroller $u(t)$ diubah pada laju proporsional dari sinyal pembangkit kesalahan $e(t)$, sehingga

$$\frac{du(t)}{dt} = K_i e(t)$$

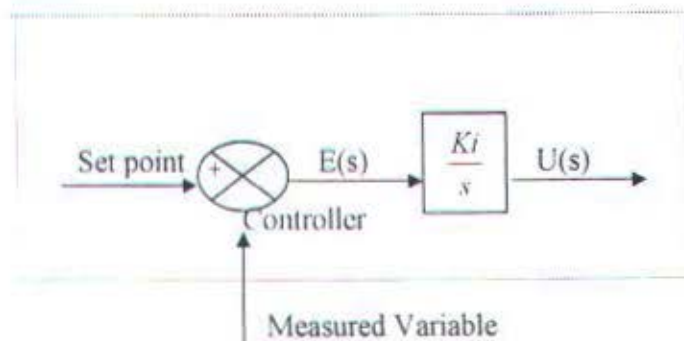
Atau

$$u(t) = K_i \int e(t) dt$$

Dengan K_i adalah konstanta yang diubah. Fungsi alih dari kontroller integral adalah (Ogata, 1997) :

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (3.7)$$

Bentuk transfer function aksi kontrol integral adalah sebagai berikut :



Gambar 1.9. Pengendali Integral

II.5.4. Pengendali Proporsional ditambah Integral (PI)

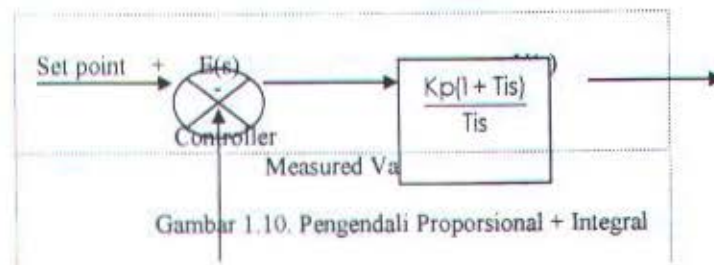
Aksi kontrol kontroller proporsional ditambah integral didefinisikan dengan persamaan berikut (Ogata, 1997) :

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int e(t) dt$$

Atau fungsi alih kontroler ini adalah :

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i(s)} \right) \quad (3.8)$$

Dengan K_p penguatan proporsional dan T_i disebut waktu integral. Keduanya K_p dan T_i dapat ditentukan. Waktu integral mengatur aksi kontrol integral sedangkan perubahan nilai K_p berakibat pada bagian aksi kontrol proporsional maupun integral. Kebalikan waktu integral T_i disebut laju reset. Laju reset adalah bilangan yang menunjukkan berapa kali tiap menit bagian proporsional dari aksi kontrol diduplikasi. Laju reset diukur dalam pengulangan per menit. Gambar 3.11 dibawah ini menunjukkan diagram blok kontroler proporsional ditambah integral.



Gambar 1.10. Pengendali Proporsional + Integral

II.5.5. Pengendali Proporsional ditambah Integral ditambah Turunan (PID)

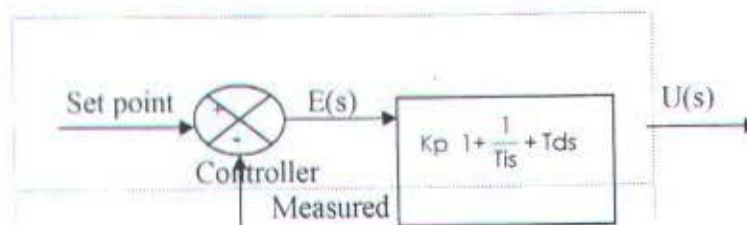
Kombinasi dari aksi kontrol proporsional, aksi kontrol integral dan aksi kontrol turunan disebut aksi kontrol proporsional ditambah integral ditambah turunan. Kombinasi ini mempunyai keuntungan dibanding masing-masing kontroler. Persamaan dengan tiga kombinasi ini adalah sebagai berikut (Ogata, 1997) :

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int e(t) dt + K_p T_d \frac{de(t)}{dt}$$

Atau fungsi alihnya

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3.9)$$

Dengan K_p penguatan proporsional, T_i waktu integral dan T_d waktu turunan. Diagram blok kontroler proporsional ditambah integral ditambah turunan ditunjukkan pada Gambar dibawah ini :



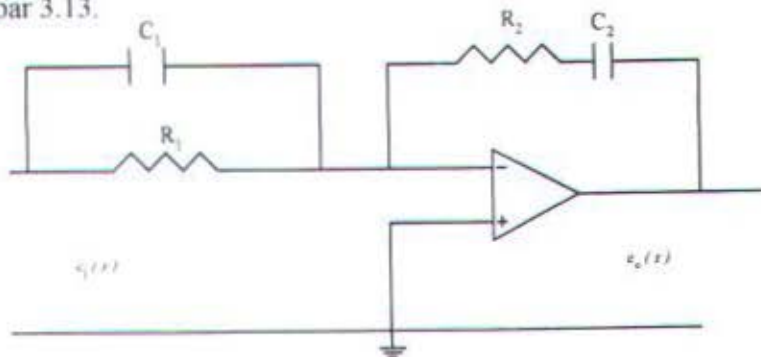
Gambar 1.11. Pengendali proporsional ditambah integral ditambah turunan

II.6. Kompensator

Kompensator digunakan didalam sistem loop tertutup untuk meningkatkan unjuk kerja system. Beberapa jenis kompensator yang digunakan adalah lag compensator, lead compensator, lag-lead kompensator. Kompensator lag merupakan bentuk rangkaian jenis proporsional-integral yang fase keluarannya tertinggal dari fase masuknya jika dikenakan sinyal masukan sinusoidal. Kompensator lead juga merupakan kompensator proporsional-integral yang fase keluarannya mendahului masukannya, sedangkan pada jenis lag-lead, fase masukannya untuk frekuensi sinyal masukan tertentu mendahului masukannya dan untuk frekuensi masukan lain bisa tertinggal dari sinyal masukan sinusoidal.

Pada rangkaian kontroler akan digunakan rangkaian elektronik dengan menggunakan penguat operasional (op – amp). Penguat operasional sering

digunakan pada sistem elektrik sebagai bagian yang penting dari pendeteksi kesalahan. Konfigurasi secara umum dari penguat operasional terlihat seperti Gambar 3.13.



Gambar 1.12. Rangkaian penguat operasional.

Pada rangkaian diatas C_1 dan C_2 dihilangkan, maka menjadi kompensator proporsional dengan nilai penguatan sebagai berikut (Ogata, 1997)

$$Kp = \frac{R_2}{R_1} \quad (3.10)$$

Rangkaian akan menjadi kompensator P_1 jika C_1 dihilangkan, sehingga didapatkan transfer function sistem adalah :

$$\begin{aligned} \frac{e_0(s)}{e_1(s)} &= \frac{R_2}{R_1} \frac{R_2 C_2 s + 1}{R_2 C_2 s} \\ &= \frac{R_2}{R_1} \left(1 + \frac{1}{R_2 C_2 s} \right) \\ &= \frac{R_2}{R_1} + \frac{1}{R_1 C_2 s} \end{aligned} \quad (3.11).$$

Didapatkan :

$$Kp = \frac{R_2}{R_1}$$

$$K_i = \frac{1}{R_1 C_2}$$

Rangkaian tersebut akan menjadi kompensator PID, jika tidak ada komponen yang dihilangkan, sehingga didapatkan transfer function sistem adalah :

$$\begin{aligned} \frac{e_o(s)}{e_i(s)} &= \frac{R_2 (R_1 C_1 s + 1)(R_2 C_2 s + 1)}{R_1 R_2 C_2 s} \\ &= \frac{R_2}{R_1} \frac{R_1 C_1 + R_2 C_2}{R_2 C_2 s} + \frac{1}{R_2 C_2 s} + R_1 C_1 s \\ &= \frac{R_1 C_1 + R_2 C_2}{R_1 C_2} \left(1 + \frac{1}{(R_1 C_1 + R_2 C_2)s} + \frac{R_1 C_1 R_2 C_2}{R_1 C_1 + R_2 C_2} s \right) \quad (3.12) \end{aligned}$$

Didapatkan : $K_p = \frac{R_1 C_1 + R_2 C_2}{R_1 C_2}$

$$T_i = R_1 C_1 + R_2 C_2$$

$$T_d = \frac{R_1 C_1 R_2 C_2}{R_1 C_1 + R_2 C_2}$$

Dinyatakan dalam penguatan proporsional, penguatan integral, dan penguatan turunan, diperoleh :

$$K_p = \frac{R_1 C_1 + R_2 C_2}{R_1 C_2}$$

$$K_i = \frac{1}{R_1 C_2}$$

$$K_d = R_2 C_1$$

II.7. Respon Transien.

Dalam beberapa kasus praktek, karakteristik kinerja yang diinginkan dari suatu system kontrol dicirikan oleh suku besaran daerah asal waktu. System dengan tenaga tidak dapat memberikan tanggapan seketika dan akan menunjukkan tanggapan transien walaupun diberi masukan atau gangguan.

Biasanya karakteristik kinerja suatu system kontrol dicirikan oleh suku tanggapan transien terhadap masukan tangga satuan karena hal itu mudah ditimbulkan dan cukup drastis. Jika tanggapan terhadap masukan tangga diketahui, secara matematis dapat dihitung tanggapan untuk sembarang masukan. Tanggapan transien suatu system terhadap masukan tangga satuan terhadap variasi system, terhadap cara praktis yang biasa dilakukan. Yaitu dengan syarat awal standart bahwa system mula-mula diam dengan keluaran dan semua turunan waktunya nol. Dengan demikian karakteristiknya dapat dengan mudah dibandingkan.

Tanggapan transien suatu system kontrol secara praktek selalu menunjukkan osilasi terendam sebelum mencapai keadaan tunaknya. Dalam menggolongkan karakteristik tanggapan transien suatu system kontrol terhadap masukan tangga satuan, umumnya dikelompokkan sebagai berikut :

1. Waktu tunda, t_d .
2. Waktu naik, t_r .
3. Waktu puncak, t_p .
4. Overshoot maksimum, M_p .
5. Waktu turun, t_s .

Pengelompokkan ini didefinisikan sebagai berikut dan secara grafik ditunjukkan pada gambar 1.13

1. Waktu tunda, t_d : Waktu tunda adalah waktu yang diperlukan oleh tanggapan untuk mencapai setengah nilai akhir untuk waktu yang pertama.
2. Waktu naik, t_r : Waktu naik adalah waktu yang diperlukan oleh tanggapan untuk naik dari 10 % menjadi 90 %, 5 % menjadi 95 %, atau 0 % menjadi 100 %, dari naik akhir yang biasa digunakan. Untuk system atas redaman waktu naik yang biasa digunakan 10 % menjadi 90 %.
3. Waktu puncak, t_p : Waktu puncak adalah waktu yang diperlukan tanggapan untuk mencapai puncak pertama overshoot.
4. Maksimum (persen) overshoot, M_p : Overshoot maksimum adalah nilai puncak kurva tanggapan diukur dari satuan. Apabila nilai akhir keadaan tunak tanggapannya jauh dari satu, maka biasa digunakan persen overshoot maksimum, dan didefinisikan oleh :

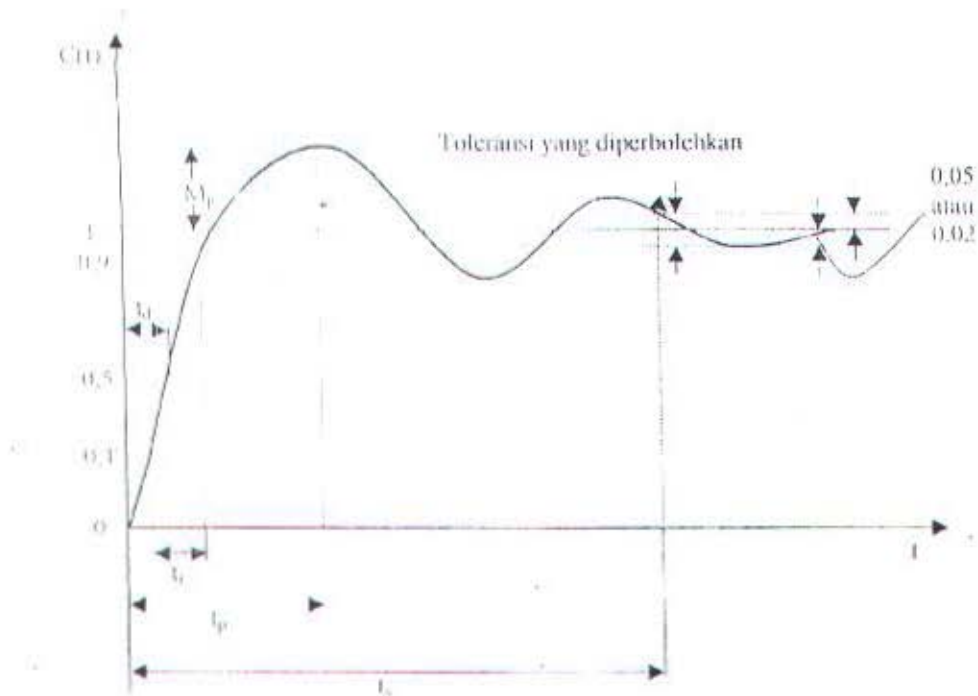
$$\text{Maksimum (persen) overshoot} = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\%$$

Besarnya persen overshoot maksimum menunjukkan kestabilan relatif dari system.

5. Waktu turun, t_s : Waktu turun adalah waktu yang diperlukan untuk menanggapi kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang disederhanakan dengan persentase mutlak harga

akhirnya (biasanya 2 % atau 5 %). Waktu turun tadi dihubungkan dengan tetapan waktu terbesar system kontrol.

Spesifikasi daerah waktu menjadi penting karena kebanyakan system kontrol merupakan system daerah waktu yaitu dapat menerima tanggapan waktu. (ini berarti bahwa system kontrol harus dimodifikasi sampai tanggapan transien dipenuhi). Perhatikan bahwa kita cirikan nilai t_d , t_r , t_p , t_s , dan M_p , maka bentuk kurva tanggapan dapat ditentukan. Dapat dilihat dengan jelas pada gambar 3.14.



Gambar 1.13

1.13 Kurva respon tangga satuan yang menunjukkan t_d , t_r , t_p , M_p dan t_s

II.8. Kriteria Kestabilan.

II.8.1. Kriteria Kestabilan Routh.

Masalah terpenting dalam sistem kontrol linier berhubungan dengan kestabilan. Dalam kondisi seperti apa sistem tidak stabil ?, apabila tidak stabil apa yang harus distabilkan dalam sistem itu ?. suatu sistem kontrol dikatakan stabil jika dia hanya semua kutub loop tertutup berada pada setengah sebelah kiri bidang s . karena sebagian besar sistem loop tertutup linier mempunyai fungsi alih loop tertutup adalah sebagai berikut (Ogata, 1997).

$$\frac{C(s)}{R(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} = \frac{B(s)}{A(s)} \quad (3.13).$$

Dengan a dan b adalah tetapan dan $m \leq n$, pertama-tama kita harus memfaktorkan polynomial $A(s)$ untuk memperoleh kutub loop tertutup. Proses ini sangat memakan waktu polynomial derajat dua atau lebih. Suatu kriteria sederhana yang disebut kriteria routh memungkinkan kita untuk menentukan jumlah kutub loop tertutup yang berada pada setengah sebelah kanan bidang s tanpa harus memfaktorkan polynomial.

Kriteria kestabilan routh memberikan informasi pada kita apakah terdapat akar positif pada persamaan polynomial tanpa penyelesaian persamaan polynomial tersebut. Apabila kriteria ini diterapkan untuk suatu sistem kontrol, informasi tentang kestabilan mutlak dapat diperoleh secara langsung dari koefisien persamaan karakteristik.

Prosedur kriteria kestabilan Routh adalah sebagai berikut :

1. Tulis persamaan polynomial dalam bidang s sebagai berikut :

$$a_n s^n = a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0 \quad (3.14)$$

Dengan koefisien merupakan besaran real. Anggap bahwa $a_n \neq 0$, sehingga terdapat akar nol yang dihilangkan.

2. Apabila terdapat koefisien nol atau negatif, maka koefisien positif terkecil adalah nol.

Akar imajiner yang mempunyai bagian real positif, dalam hal ini *sistem tidak stabil*.

3. Jika semua koefisien, susun koefisien polynomial dalam baris kolom sesuai pada pola berikut :

s^n	a_0	a_2	a_4	a_6	\dots
s^{n-1}	a_1	a_3	a_5	a_7	\dots
s^{n-2}	b_1	b_2	b_3	b_4	\dots
s^{n-3}	c_1	c_2	c_3	c_4	\dots
s^{n-4}	d_1	d_2	d_3	d_4	\dots
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
s_2	e_1	e_2			
s_1	f_1				
s_0	g_1				

dimana :

$$b_1 = \frac{a_1 a_2 - a_0 a_3}{a_1}$$

$$c_1 = \frac{b_1 a_3 - a_1 b_2}{b_1}$$

$$d_1 = \frac{c_1 b_2 - b_1 c_2}{c_1}$$

$$b_2 = \frac{a_1 a_4 - a_0 a_5}{a_1}$$

$$c_2 = \frac{b_1 a_5 - a_1 b_3}{b_1}$$

$$d_2 = \frac{c_1 b_3 - b_1 c_3}{c_1}$$

$$b_3 = \frac{a_1 a_6 - a_0 a_7}{a_1}$$

$$c_3 = \frac{b_1 a_7 - a_1 b_4}{b_1}$$

Proses ini diteruskan sampai baris ke-n secara lengkap.

Kriteria kestabilan Routh menyatakan bahwa jumlah akar persamaan 3. 15 dengan bagian real positif sama dengan jumlah perubahan data dari koefisien kolom pertama. Harus diperhatikan bahwa nilai yang tepat pada kolom pertama tidak dipentingkan, hanya perubahan tanda yang harus diperhatikan. Syarat perlu dan syarat cukup agar sistem stabil, adalah semua koefisien pada kolom pertama mempunyai tanda positif.

II.8.2. Kriteria Kestabilan Root-locus.

Suatu metode sederhana untuk mencari akar-akar persamaan karakteristik telah ditemukan oleh W. R. Evans dan digunakan secara luas dalam teknik kontrol. Metode ini, yang disebut metode *tempat kedudukan akar*, merupakan suatu metode dengan menggambarkan akar-akar persamaan karakteristik untuk semua nilai dari suatu parameter sistem. Akar-akar untuk suatu nilai tertentu dari parameter selanjutnya terletak pada grafik yang diperoleh. Perhatikan bahwa parameter ini biasanya adalah penguatan, tetapi setiap variabel lain dari fungsi ahli loop terbuka juga dapat digunakan. Jika tidak disebutkan, kita anggap bahwa peringatan fungsi loop terbuka merupakan parameter yang akan diubah di seluruh daerah harganya yakni dari nol sampai tak terhingga.

Dengan menggunakan metode tempat kedudukan akar, perancang dapat memprediksikan pengaruh ada lokasi kutub loop tertutup jika nilai penguatan bervariasi atau penambahan kutub loop terbuka dan atau nol kutub terbuka. Oleh

karena itu, hal yang diinginkan bahwa perancang memiliki pengertian yang baik tentang metode untuk menata tempat kedudukan akar sistem loop tertutup.

Metode tempat kedudukan akar. Gagasan dasar dari metode tempat kedudukan akar adalah harga s yang membuat fungsi ahli loop tertutup jika penguatan diubah dari nol sampai tak terhingga memberikan latar belakang pemberian nama metode ini. Diagram ini secara jelas menunjukkan kontribusi tiap kutub atau zero loop terbuka terhadap letak kutub-kutub loop tertutup.

Metode tempat kedudukan akar memungkinkan kita untuk mencari kutub loop tertutup dari kutub dan zero loop terbuka dengan penguatan sebagai parameter. Metode ini menghilangkan kesulitan/kesulitan yang timbul pada teknik klasik dengan memberikan peragaan grafis semua kutub loop tertutup untuk semua harga penguatan fungsi ahli loop terbuka. Dalam mendesain sistem kontrol linier, metode tempat kedudukan akan terbukti cukup berguna karena menunjukkan cara memodifikasi kutub dan zero loop terbuka sedemikian rupa sehingga respon memenuhi spesifikasi unjuk kerja sistem. Metode ini sangat cocok untuk mencari hasil pendekatan secara tepat. Karena metode ini memberikan suatu prosedur grafis yang efektif untuk mencari akar-akar setiap polynomial yang timbul dalam studi sistem fisik.

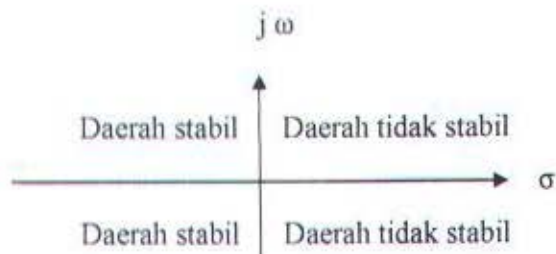
Beberapa sistem kontrol mungkin lebih dari satu parameter yang harus diatur, maka diagram tempat kedudukan akar untuk sistem yang memiliki banyak parameter mungkin dibentuk dengan variasi satu parameter terhadap waktu. Dalam bab ini kita membicarakan tempat kedudukan akar untuk sistem yang



memiliki dua parameter. Tempat kedudukan akar untuk kasus semacam ini dikatakan sebagai *root contour* (kontur akar).

Daerah kestabilan dalam bidang $-s$ (bidang kompleks).

Kedudukan akar-akar karakteristik dari persamaan karakteristik suatu sistem dalam bidang $-s$ kompleks ($s = \sigma + j\omega$), bisa digunakan untuk mengetahui apakah sistem stabil atau tidak stabil. Gambar dibawah ini menunjukkan daerah kestabilan sistem berdasarkan tempat kedudukan akar-akar.



Gambar 3. 15 daerah kestabilan dalam bidang kompleks.

BAB III

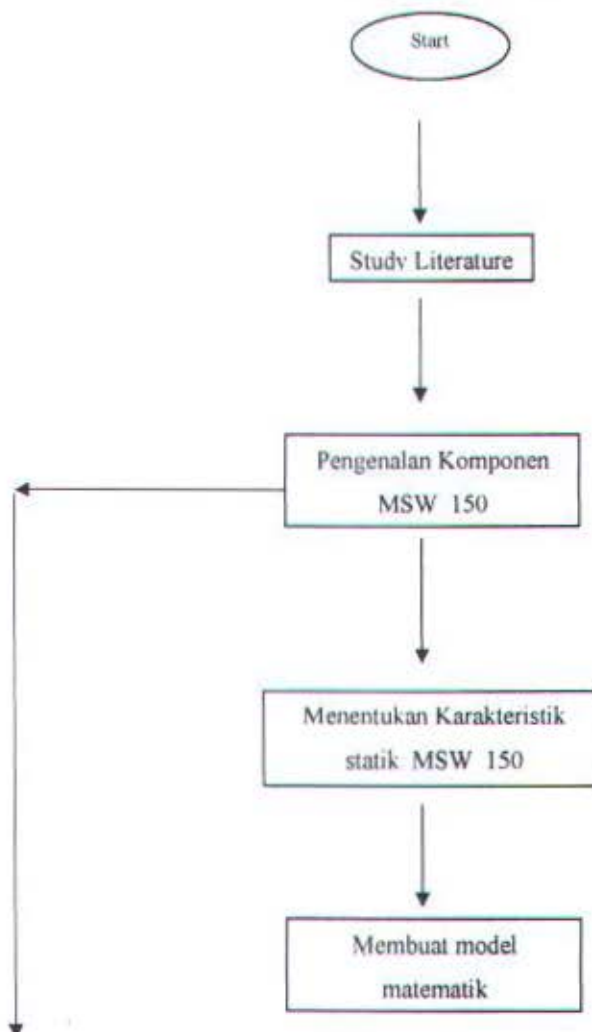
METODOLOGI

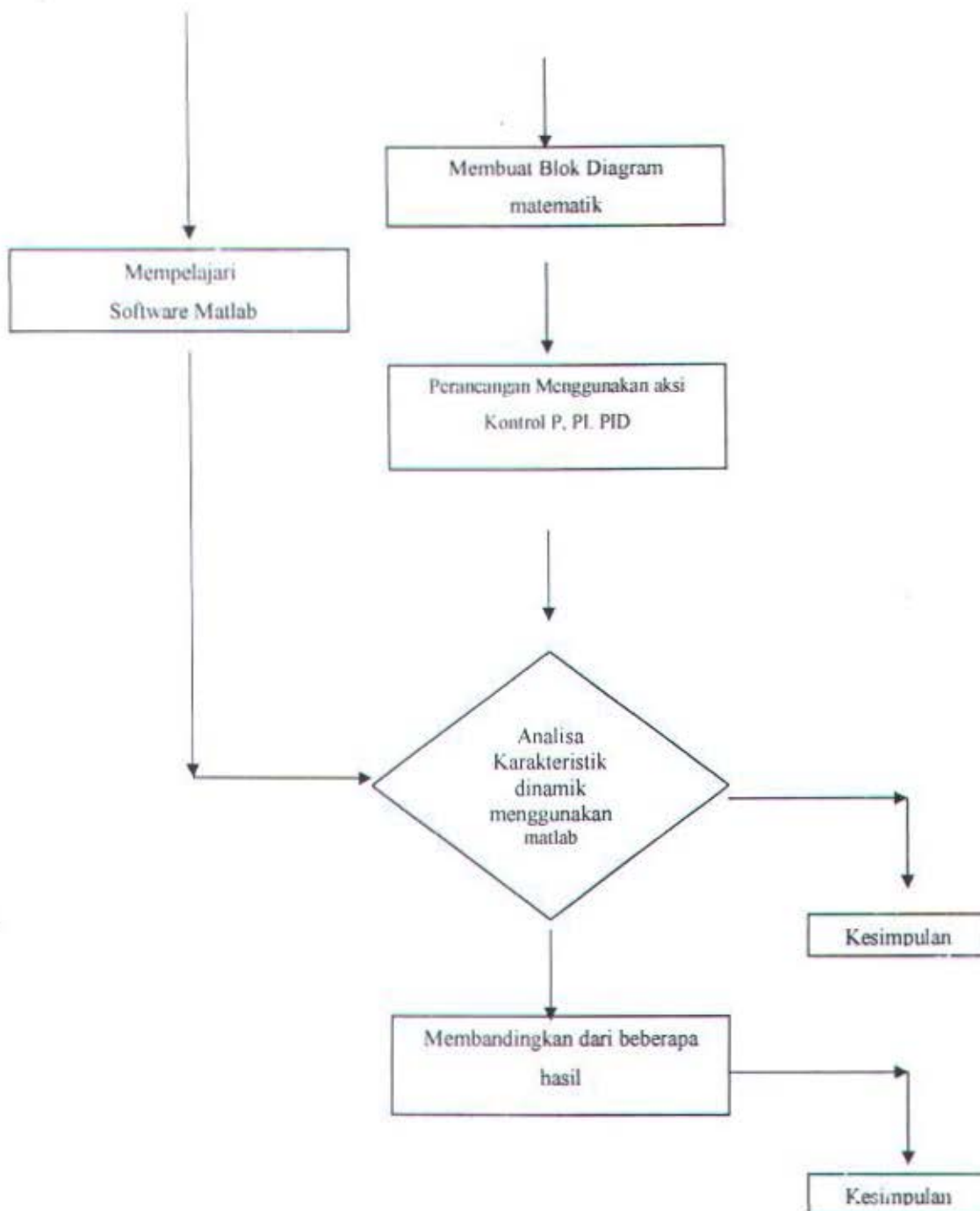
Metode yang akan digunakan dalam penulisan tugas akhir ini adalah dengan metode eksperimen dengan urutan kerja sebagai berikut :

1. Studi literatur dengan mempelajari teori – teori penunjang yang diperoleh dari literatur terkait misalnya model dinamis simulasi ,sistem pengendalian,dll
2. Sistem kontrol harus memiliki spesifikasi sebagai berikut:dapat memperkecil kesalahan sampai nol atau pada suatu nilai dapat ditoleransi, memiliki kecepatan respon cukup cepat dan menunjukkan peredaman yang layak dan kestabilan relatif yang layak .
3. Mempelajari instrumen yang terdapat pada MSW 150 dan menentukan karakteristik statik instrumen tersebut
4. Menurunkan model matematis dari komponen sistem yang kemudian dibawa ke bentuk laplace.
5. Membuat blok diagram dari alat kontrol MSW 150
6. Menerapkan aksi kontrol yaitu Kontrol Proportional (P) , Kontrol Proportional plus Integral (PI),Kontrol Proportional plus Integral plus Derivativ (PID).
7. Menganalisa Karakteristik dinamik dari tiap – tiap aksi kontrol menggunakan software matlab

8. Melihat apakah aksi kontrol tersebut sudah memenuhi spesifikasi ,apabila masih belum memenuhi spesifikasi maka pengujian dilakukan kembali sampai spesifikasi terpenuhi
9. Apabila sudah memenuhi spesifikasi maka akan diketahui aksi kontrol yang memberikan hasil paling optimal pada alat kontrol MSW 150

Untuk lebih memudahkan dalam memahami metode yang akan digunakan, berikut ini diberikan diagram alir untuk penyelesaian tugas akhir

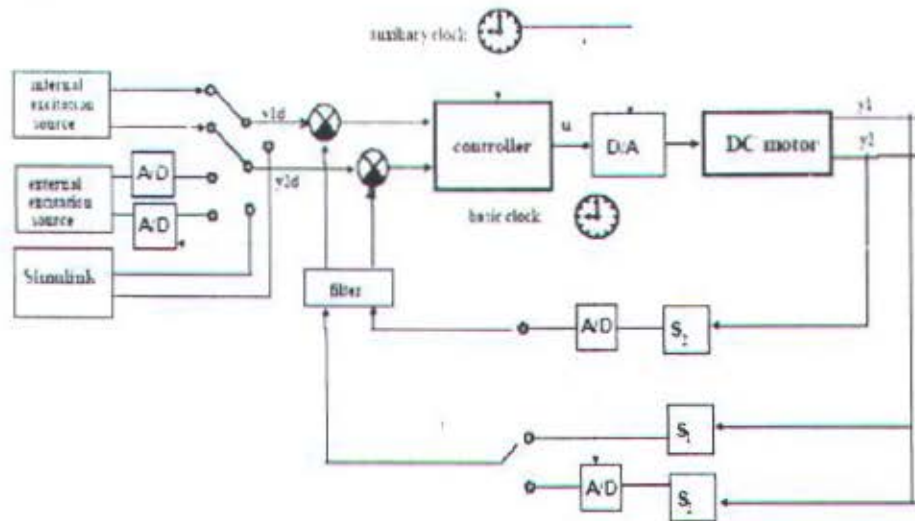




BAB IV

PENGEMBANGAN MODEL MATEMATIS DAN BLOK DIAGRAM MSW 150

Dari manual book reference MSW 150 didapatkan basic block diagram sebagai berikut:



Dimana :

y1 = posisi

y2 = kecepatan

S1 = Potensiometer untuk posisi, Tachogenerator untuk kecepatan.

S2 = Encoder untuk posisi

Internal Excitation Source = RTK

External Excitation Source = Modular Servo Workshop 150

A/D dan D/A Converter = Papan Data Akusisi

Simulink Excitation Source = Dari Mathworks.Inc Software.

Function untuk main control window dari MSW 150

```
function Es_model()
```

```
% This is the machine-generated representation of a Handle Graphics object
% and its children. Note that handle values may change when these objects
% are re-created. This may cause problems with any callbacks written to
% depend on the value of the handle at the time the object was saved.
%
```

% To reopen this object, just type the name of the M-file at the MATLAB
 % prompt. The M-file and its associated MAT-file must be on your path.

```
load Es_model
```

```
a = figure('Units','normalized', ...
    'Color',[0.45 0.66 0.143], ...
    'Colormap',mat0, ...
    'CreateFcn','IfOpWin("Set build-in model parameters");', ...
    'MenuBar','none', ...
    'Name','Set build-in model parameters', ...
    'NumberTitle','off', ...
    'PointerShapeCDData',mat1, ...
    'Position',[0.21875 0.345 0.55125 0.343333], ...
    'Tag','Fig1');
b = uicontrol('Parent',a, ...
    'Units','normalized', ...
    'BackgroundColor',[0.5 0.5 0.5], ...
    'Position',[0.127272 0.20155 0.445455 0.654971], ...
    'Style','frame', ...
    'Tag','Frame1');
b = uicontrol('Parent',a, ...
    'Units','normalized', ...
    'BackgroundColor',[0.5 0.5 0.5], ...
    'FontSize',10, ...
    'FontWeight','demi', ...
    'ForegroundColor',[1 1 1], ...
    'Position',[0.179545 0.58 0.136364 0.111111], ...
    'String','Ks :', ...
    'Style','text', ...
    'Tag','StaticText1');
b = uicontrol('Parent',a, ...
    'Units','normalized', ...
    'BackgroundColor',[0.5 0.5 0.5], ...
    'FontSize',10, ...
    'FontWeight','demi', ...
    'ForegroundColor',[1 1 1], ...
    'Position',[0.179545 0.32 0.136364 0.116959], ...
    'String','Ts :', ...
    'Style','text', ...
    'Tag','StaticText2');
b = uicontrol('Parent',a, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'CreateFcn','rem = es_call("getmodelp");rem = num2str(rem(1));set(gcbo,
    "String",rem);', ...
```

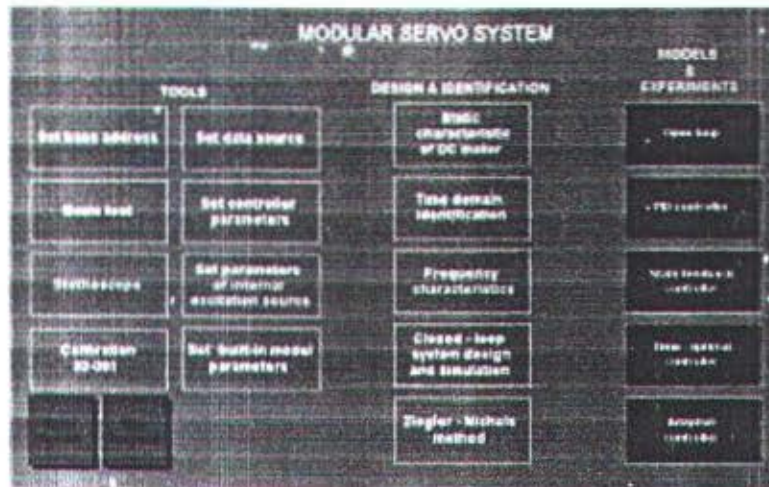


```

'FontSize',10, ...
'FontWeight','demi', ...
'HorizontalAlignment','right', ...
'Position',[0.352273 0.6 0.181818 0.116959], ...
'String','230', ...
'Style','edit', ...
'Tag','EditKs');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'CreateFcn',rem = es_call("getmodelp");rem = num2str(rem(2));set(gcbo,
"String",rem), ...
'FontSize',10, ...
'FontWeight','demi', ...
'HorizontalAlignment','right', ...
'Position',[0.352273 0.35 0.181818 0.116959], ...
'String','0.11', ...
'Style','edit', ...
'Tag','EditTs');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'BackgroundColor',[0.5 0.5 0.5], ...
'Position',[0.629545 0.20155 0.247727 0.651163], ...
'Style','frame', ...
'Tag','Frame2');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback',mat2, ...
'FontSize',10, ...
'FontWeight','demi', ...
'Position',[0.656818 0.668605 0.195455 0.127907], ...
'String','SET', ...
'Tag','Pushbutton1');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback','helpwin es_mdl_hlp', ...
'FontSize',10, ...
'FontWeight','demi', ...
'Position',[0.656818 0.459302 0.195455 0.127907], ...
'String','HELP', ...
'Tag','Pushbutton2');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback','close;', ...
'FontSize',10, ...
'FontWeight','demi', ...

```

```
'Position',[0.656818 0.255814 0.195455 0.127907], ...
'String','CLOSE', ...
'Tag','Pushbutton3');
```



Gbr. main control window MSW 150

Function untuk percobaan PID controller

```
function [sys, x0, str, ts] = sfunc( t, x, u, flag, Par, T0, ds, exc_src, SetRTKPar,
flag9 )
%
% S-function pid_sf.M - S-function for PID controller
%
% Parameters:
% t      - time,
% x      - state vector,
% u      - input to the S-function block,
% flag   - the value passed to S-function by SIMULINK to distinguished
%         different actions. The arguments t, x, u and flag are set and
%         passed to S-function by SIMULINK automatically,
% downsamp - downsampling ratio. Defines how many samples is transfered to
%         the output of the S-function block. If downsamp is equal
%         to 10 only 1 sample of every 10 samples is transfered from
%         Real Time Task to the output of the S-function block.
% exc_src - simulink generator flag. If this value is equal to two
%         simulink generator sets desired position of the cart. The
%         simulink generator is connected to the input of the S-function
```

```

%      block.
% stop_pract - stop practical after simulation/stop

% Global variables
global history es_flag9
global local_downsamp

switch flag

case 0, % Initialization
    % Set number of continuous states, number of discrete states, number of
    % outputs and number of inputs.
    % 0 continuous states, 1 discrete state, 7 outputs, 1 inputs
    sizes.NumContStates = 0;
    sizes.NumDiscStates = 1;
    sizes.NumOutputs = 7;
    sizes.NumInputs = 1;
    sizes.DirFeedthrough = 0;
    sizes.NumSampleTimes = 1;
    sys = simsizes(sizes);

    local_downsamp = ds;
    es_flag9 = flag9;

    % Set initial values of the control algorithm
    if ( eq( SetRTKPar, 1 ) )
        dummy = es_call( 'SetAlgNo', 0 );
        dummy = es_call( 'SetSampleTime', T0 );
        dummy = es_call( 'SetP', Par );
        dummy = es_call( 'SetInitCond', [0 0] );
        dummy = es_call( 'SetAlgNo', 2 ); % PID
        dummy = es_call( 'ResetTime', 0 );
        p = es_call( 'GetPW' );
        p( 1 : 2 ) = [ 0 0 ];
        p = es_call( 'SetPW', p );

    switch exc_src,
        case 1, % Simulink signal generator
            dummy = es_call( 'SetDataSource', [ 8 9 13 4 5 6 ] );
            dummy = es_call( 'SetPW', [ 0 0 zeros( 1, 18 ) ] );
        case 2, % External signal source
            dummy = es_call( 'SetDataSource', [ 8 9 12 4 5 6 ] );
        case 3, % Built-in signal generator (square)
            dummy = es_call( 'SetDataSource', [ 8 9 13 4 5 6 ] );
            dummy = es_call( 'SetPW', [ 1 3 3 2 2 -50 50 -50 50 zeros( 1, 11 ) ] );
    end
end

```



```

es_cpar( 'esc_UpdateButton' );
es_egen( 'esg_UpdateButton' );

end

    % Wait for the first sample which may be send to the output
    while ( es_call( 'GetNoOfSamples', 0 ) <= ds )
        ;
    end;

history = es_call( 'GetHistory', 0 );

% initialize the initial conditions
str = []; % str is always an empty matrix
ts = [-2 0]; % initialize the array of sample times
    % variable sample time
    % Set initial conditions of the state
    x0 = 1;

% change background color of Real Time Task block after simulation start
set_param('es_pid/RealTimeTask','BackgroundColor','cyan');

case 1, % Unhandled flags
    sys = [];

case 2, % Calculate discrete state
    % Set desired value
    if eq( exc_src, 1 ) % Simulink signal generator
        dummy = es_call( 'SetPW', [ 0 u zeros( 1, 18 ) ] );
    end
    sys = x;

case 3, % Calculate outputs
    [ sys, history ] = es_sfco( local_downsamp, history );

case 4, % Calculate next discrete time point
    [ sys, local_downsamp, history ] = es_sfntp( local_downsamp, history );

case 9, % Terminate
    % change background color of Real Time Task block after simulation stop
    set_param('es_pid/RealTimeTask','BackgroundColor','green');

    if ( eq( es_flag9, 1 ) )
        dummy = es_call( 'SetAlgNo', 0 );
        dummy = es_call( 'SetSampleTime', 0.05 );
    end

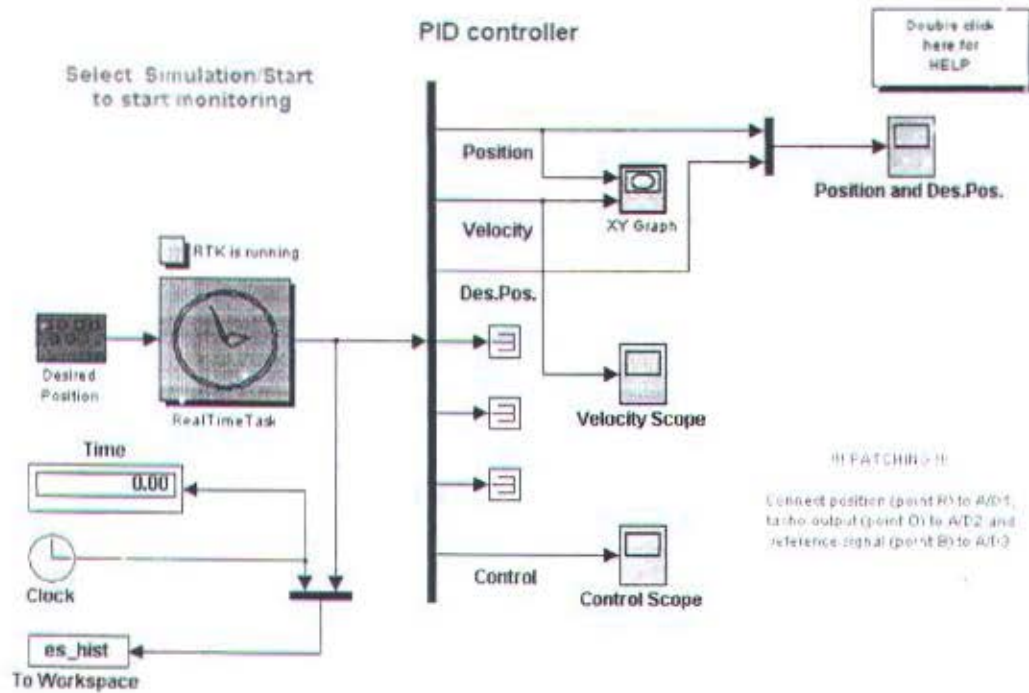
```

```

end

otherwise % Unexpected flags %
    error(['Unexpected flag = ', num2str(flag)]);
end

```



Gbr.PID Controlled window

Parameter – parameter dalam PID

BAB V

ANALISA DATA DAN PEMBAHASAN

Dalam menganalisa system kontrol diperlukan untuk mengetahui karakteristik static dari komponen yang digunakan :

5.1. Operational Amplifier

1.1 Efek Penjumlahan pada Operational Amplifier (OP-Amp)

Tujuan :

- Mengetahui bagaimana Operational Amplifier (OP-Amp) dapat digunakan untuk menjumlah input.
- Dapat menggunakan Operational Amplifier (OP-Amp) untuk mengetahui skala dari input dan sebagai control gain.

Berikut ini gbr rangkaian ,

As an experiment to check the summing effect of the operational amplifier, connect up as in fig 3.2.5.

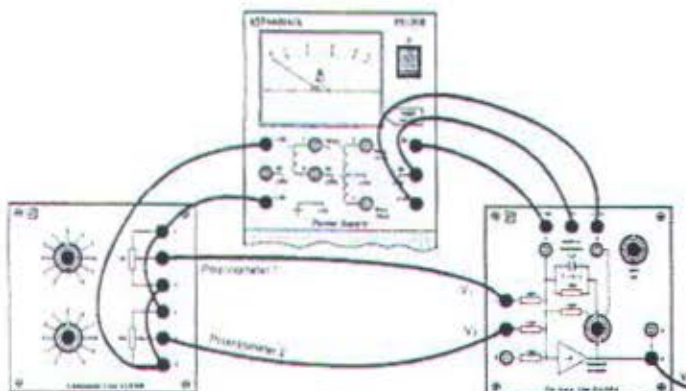


Fig 3.2.5

Langkah – langkah:

- Atur switch back selector switch pada posisi 100 k Ω resistor
- Atur arus pada terminal AU 150 B (1 & 2) dan (4&5)
- Atur output dari OA 150 B ,terminal 6 untuk berikan pembacaan nol.
- Jaga Potensiometer 1 (bagian atas) pada nol,atur potensiometer 2 pada + 2 volts
- Ukur Vo(output) dan masukkan nilainya dalam tabel.
- Atur potensiometer 2 pada variasi voltage yangg berbeda dengan tetap menjaga voltage dari potensiometer 1 pada nol.

Tabel hasil pembacaan

V1	V2	Vo
0	2	0.69
0	3	0.69
0	4	0.69
0	5	0.69
0	6	0.69

Langkah – langkah:

- Kemudian Jaga Potensiometer 2 pada nol,atur potensiometer 1 pada + 2 volts
- Ukur Vo(output) dan masukkan nilainya dalam tabel.
- Atur potensiometer 1 pada variasi voltage yangg berbeda dengan tetap menjaga voltage dari potensiometer 2 pada nol.

Tabel hasil pembacaan

V1	V2	Vo
2	0	0.69
3	0	0.69
4	0	0.69
5	0	0.69
6	0	0.69

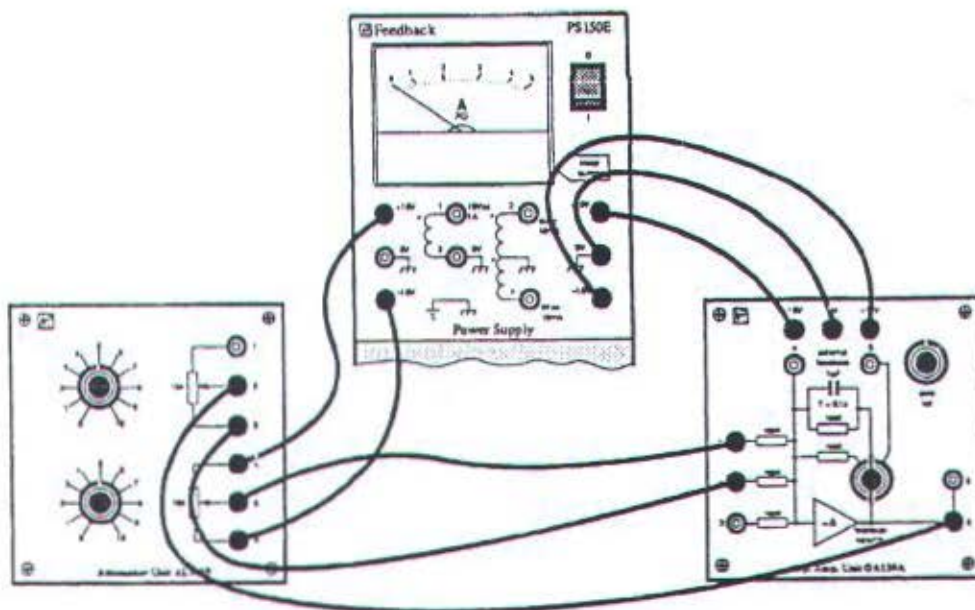
Dari hasil percobaan diatas diketahui bahwa nilai dari V_1 dan V_2 sama, yang membuktikan kalau op-amp ini dalam keadaan baik.

5.1.2 .Penyekalaan menggunakan Feedback Resistor.

Tujuan:

- Mengetahui alasan / sebab adanya perbedaan antara hasil yang diukur dengan hasil perhitungan

Gbr.Rangkaian.



Langkah – langkah :

- Atur Feedback selector switch pada posisi External Feedback
- Atur arus input + 2 volts dan ukur V_o dengan menggunakan setting 0-10 pada potensiometer.
- Ukur dan masukkan nilainya dalam tabel :

Tabel percobaan

V1	R2 setting	Vo	Vo/V1
2	0	2.75	1.375
2	10	2.75	1.375

Dari percobaan diatas dapat diketahui bahwa faktor skala V_o/V_i adalah -

1,375 volts

5.1.3 Gain Control

Tujuan:

Untuk mengetahui gain control dari Operational amplifier (OP-Amp)

Gbr rangkaian:

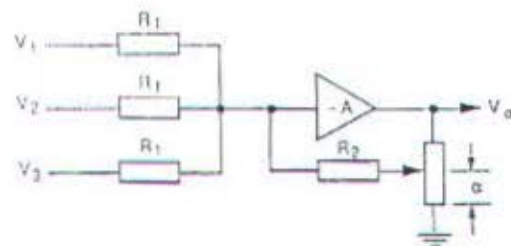
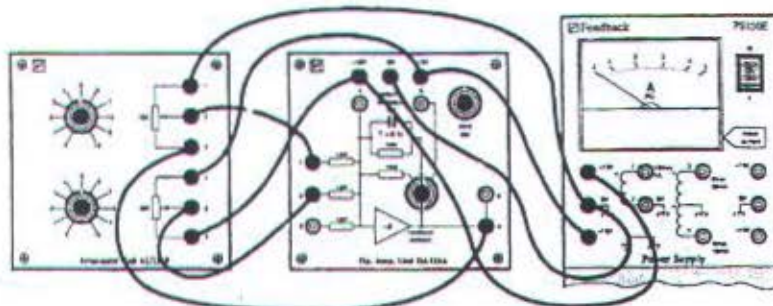


Fig 3.2.10

So set up as in fig 3.2.11 using the same units as in Practical 2.2.



Langkah – langkah :

- Atur Terminal 3 pada potensiometer Amplifier
- Atur selector switch pada External Feedback
- Hubungkan voltmeter antara V_o output dari OA150B dan 0v.
- Ukur dan masukkan dalam tabel
- Untuk posisi 1 – 10 pada potensiometer 1 sedangkan 1volts untuk potensiometer 2

Tabel

Pot 1	Pot 2 Volts	V_o Volts	gain V_o/V_1 Volts	Calculated gain
1	1	27.21	27.21	10
2	1	27.19	27.19	5
5	1	27.15	27.15	2
10	1	27.09	27.09	1
1	2	12.25	6.125	10
2	2	12.24	6.12	5
5	2	12.2	6.1	2
10	2	12.14	6.07	1

Kemudian untuk 2 volt pada potensiometer 2

Hasil:

Dari percobaan diatas setelah dibandingkan antara gain dari tiap potensiometer dapat diambil kesimpulan bahwa hasilnya berbeda pada potensiometer yang berbeda,hal ini seharusnya tidak terjadi .Kemungkinan penyebabnya adalah

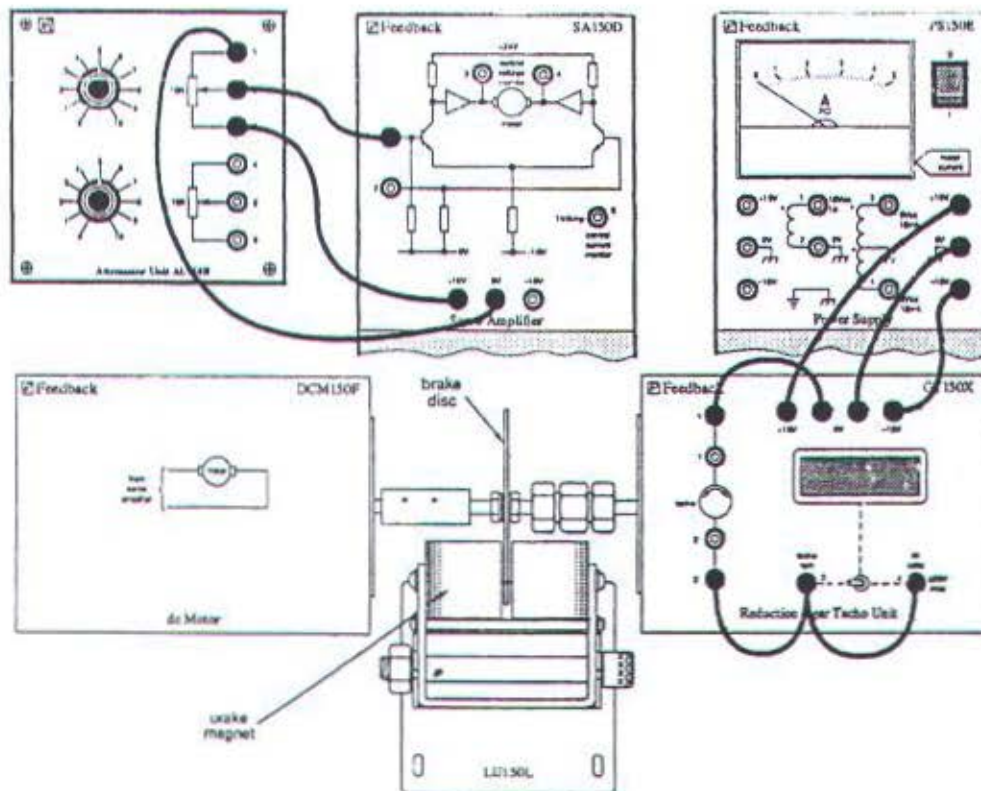
1. kesalahan baca/human error,
2. kesalahan pada voltmeter.,kalibrasi voltmeter
3. Perbedaan karena nilai resistor yang berbeda.

5.2. Characteristic Motor DC

Tujuan :

- untuk mengetahui karakteristik dari torsi/kecepatan
- mengetahui faktor skala dari tachogenerator
- mengetahui bahwa motor DC dapat dikontrol dengan memvariasikan magnitude dan beban pada pengaturan arus armatur.

Gbr Rangkaian

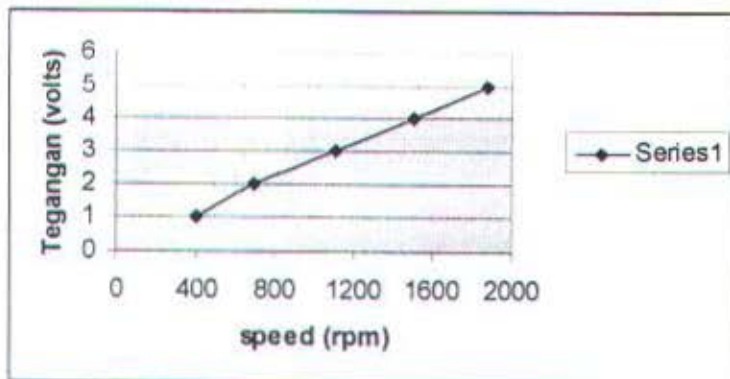


Langkah –Langkah :

- atur rem pada nol/tidak ada pengereman
- atur potencimeter sehingga terbaca 1 voltage pada Tachogenerator volts
- Variasikan potencimeter dengan voltage yang berbeda

Tabel hasil pengukuran:

Tachogenerator Volts	Speed r/min
1	400
2	700
3	1110
4	1510
5	1880



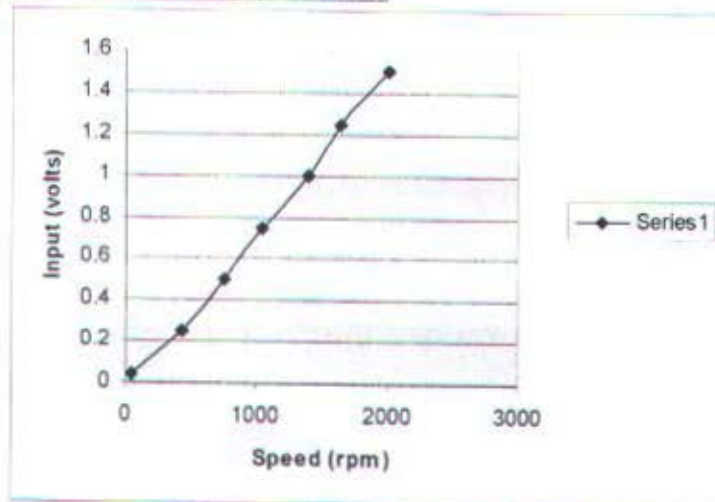
Grafik arus terhadap kecepatan.

- Dari grafik diatas dapat disimpulkan bahwa dengan semakin besar tegangan yang dialirkan maka kecepatan dari motor meningkat secara proportional terhadap kenaikan tegangan ,hal ini dibuktikan dengan kemiringan dari grafik tegangan terhadap kecepatan.
- Dari tabel hasil pengukuran dapat diketahui faktor kalibrasi yaitu $K_g = V_g/N$
 $= 3/1000 = 0,003$.

Untuk mengetahui slope/ kemiringan dari kurva kecepatan/input tanpa beban maka harus diatur agar arus input berkurang sampai motor hanya berputar. kemudian naikan voltasenya tiap 1 volts dan ukur voltage input dari potensiometer dan voltage tachogenerator sampai mendekati putaran 2000 rpm yang merupakan batas maximum dari kecepatan Motor DC ini.

Tabel hasil Pengukuran:

Vin Volts	Vg Volts	Speed Rpm
0.04	0.2	50
0.25	1.15	430
0.5	2.09	750
0.75	2.81	1040
1	3.76	1400
1.25	4.4	1640
1.5	5.31	2000



Grafik. Input terhadap kecepatan

- Dari grafik diatas dapat diketahui slope/kemiringan dari kecepatan terhadap arus input tanpa beban. ini untuk membuktikan kerja dari motor. Kenaikan arus input sebanding atau proportional dengan kenaikan dari kecepatannya.

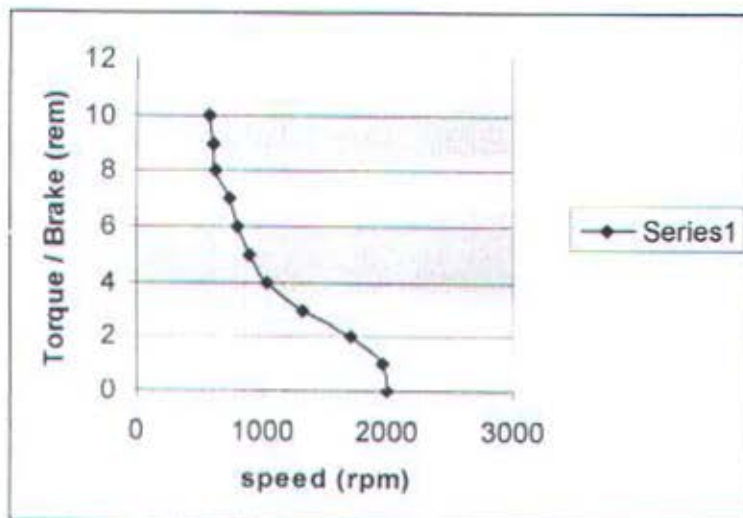
Percobaan 2.2

- Untuk mengetahui torsi/kecepatan maka digunakan beban yaitu rem.

- Atur rem pada posisi 0 dan naikan arus voltage sampai motor mendekati kecepatan maksimal
- Kemudian atur rem pada posisi maximum yaitu 10,ukur arus inputnya..
- Ukur arus dengan pengereman dari 10 – 1

Tabel hasil pengukuran

Brake Position	Vg Volts	Speed rpm
10	1.57	580
9	1.6	610
8	1.73	630
7	1.95	730
6	2.17	800
5	2.37	890
4	2.77	1040
3	3.58	1330
2	1.73	1710
1	4.57	1970
0	5.29	2000



Grafik ,beban terhadap kecepatan

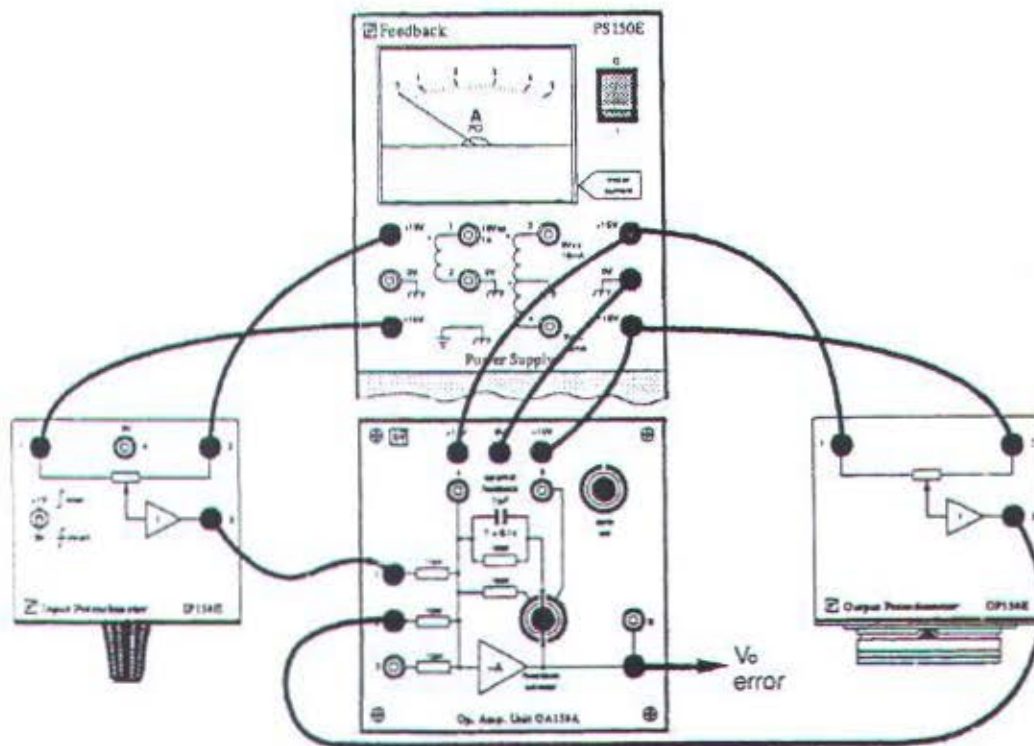
Dari grafik data diatas dapat diambil kesimpulan bahwa semakin besar putaran akan menunjukkan pada torsi yang kecil/hambatan yang ada kecil dan apabila putaran motor perlahan – lahan menunjukkan bahwa beban/torsi terhadap system yang ada besar

5.3. Error channel

Tujuan :

- Mengetahui bagaimana error signal dapat terjadi dan dapat digunakan oleh Operational amplifier Sebagai Pembanding
- Dapat Menjelaskan Error signal
- Mengetahui bagaimana 2 potensiometer dapat menghasilkan error signal

Gbr rangkaian :



Langkah – langkah :

- Atur Feedback selector switch pada 100k Ω
- Hubungkan voltmeter dengan output dari OA 150 A,atur dan setting zero set mendekati angka nol pada pembacaan voltmeter
- Atur input dan output potensiometer pada nol derajat

- Kemudian atur input dan output potensiometer pada 39 derajat
- Selanjutnya atur pada derajat yang lain

Tabel hasil percobaan

Scale Reading degree	Amplifier output Vo volts
39	-6.34
60	-10.04
90	-13.51
120	-13.49

Dari tabel diatas diketahui bahwa output yang terbaca pada amplifier tidak sama hal ini disebabkan karena kurang telitian dalam pembacaan sudut dari input dan output potensiometer, perbedaan / selisih derajat pada kedua alat tersebut akan mengubah keseimbangan yang akan mengakibatkan berubahnya nilai dari output amplifier.

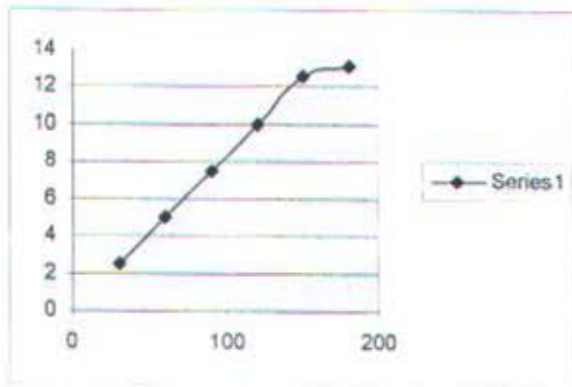
Langkah – langkah :

- atur output potensiometer pada nol derajat.
- Putar input potensiometer pada derajat tertentu

Tabel hasil percobaan:

Output Degree	Input degree	Vo Volts
0	30	2.55
0	60	4.96
0	90	7.49
0	120	9.95
0	150	12.57
0	180	13.18





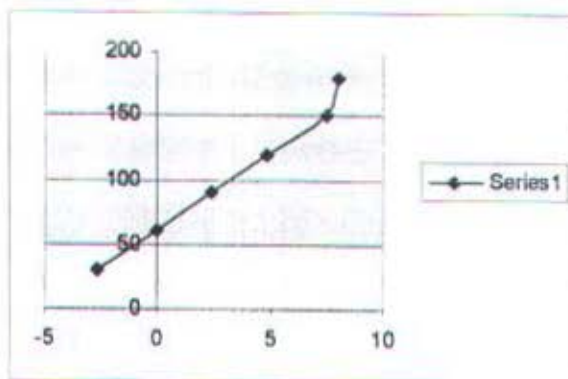
Grafik.input potensiometer terhadap output amplifier

Langkah – langkah :

- Atur output potensiometer pada - 60 derajat.

Tabel hasil percobaan

Output degree	Input degree	Vo Volts
-60	30	-2.67
-60	60	-0.05
-60	90	2.41
-60	120	4.82
-60	150	7.52
-60	180	8.08



Grafik.input potensiometer dengan amplifier output

Dari kedua tabel diatas dapat disimpulkan bahwa mengubah output dari output potensiometer akan mengubah posisi dari dari input potensiometer tapi tidak akan mengubah kemiringan dari grafik. Dari tabel diatas juga dapat diketahui tentang error channel yaitu error factor yang merupakan missaligment dari input dan output shaft. Dari perhitungan dapat diketahui bahw aerror factor:

$$K_e = \text{Volts/degrees}$$

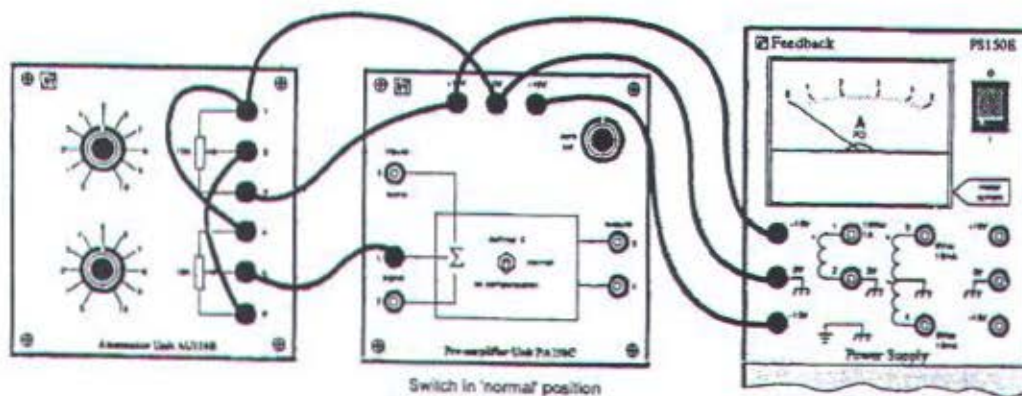
$$K_e = 0.09$$

5.4 . closed loop position control system

Tujuan :

- mengetahui karakteristik dari pre amplifier PA150 C
- mengetahui aksi closed loop control system

gbr rangkaian.



Langkah – langkah :

- atur potensiometer bagian atas pada + 2 volt

Tabel :

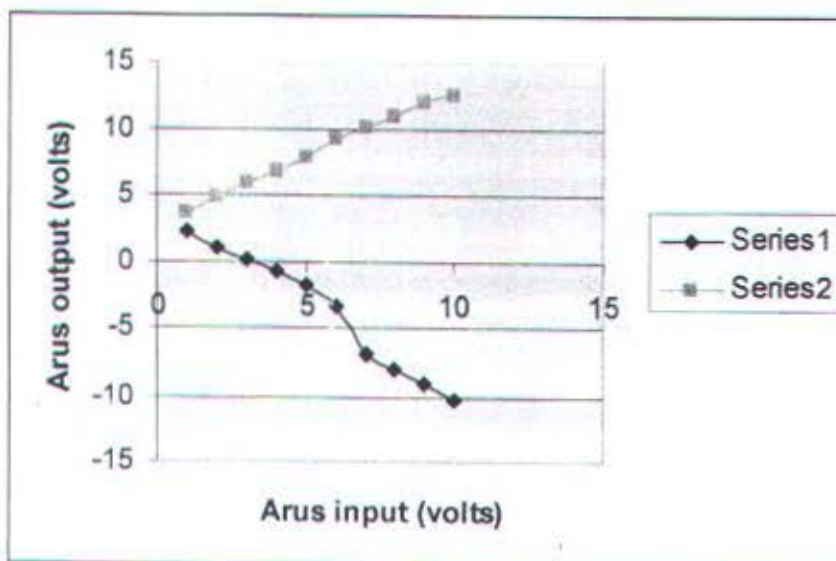
Input Signal V_i volts	Pre Amplifier Output		
	$V_o(3)$	$V_o(4)$	$V_o(4-3)$
0.06	0.95	3.15	2.2
0.19	0.95	1.93	0.98
0.28	0.97	1	0.03
0.37	1.82	0.95	-0.87

0.47	2.86	0.95	-1.91
0.6	4.34	0.95	-3.39
0.71	7.85	0.95	-6.9
0.78	8.95	0.95	-8
0.89	10.01	0.95	-9.06
1	11.17	0.95	-10.22

Kemudian hubungkan terminal 3 dengan -15 volt supply

Tabel Input sinal terhadap Preamp output.

Input Signal V_i volts	Pre Amplifier Output		
	$V_o(3)$	$V_o(4)$	$V_o(4-3)$
0.06	0.95	4.57	3.62
0.19	0.95	5.78	4.83
0.28	0.95	6.84	5.89
0.37	0.95	7.78	6.83
0.47	0.95	8.69	7.74
0.6	0.95	10.15	9.2
0.71	0.95	10.99	10.04
0.78	0.95	11.95	11
0.89	0.95	13.01	12.06
1	0.95	13.55	12.6



Grafik. Arus input terhadap Arus output.

Dari grafik diatas perbedaan antara series 1 dan 2 dikarenakan:

1. Human Error/kesalahan baca.
2. Error pada voltmeter.
3. Ketidaklinieran dimungkinkan disebabkan karena pre amplifier bekerja pada ambang batas maximal tegangan sehingga data yang didapatkan kurang valid.
4. Disebabkan karena pre amp belum di set secara tepat pada nol.

5.4.2 Simple motor driven closed loop position control system

Gbr rangkaian :

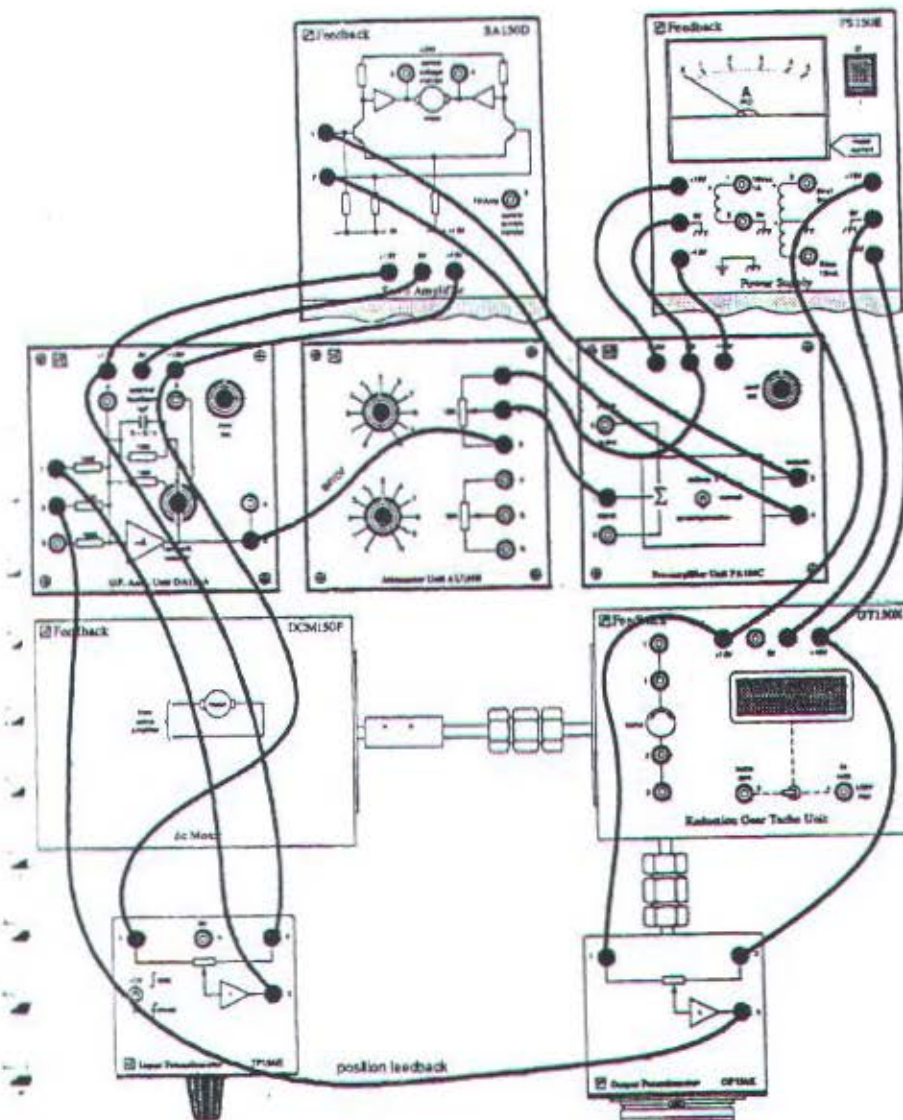


Fig 3.6.5

Langkah – langkah :

- Atur AU 150 B pada nol, agar motor tidak berputar saat dinyalakan
- Atur PA 150 C agar motor berhenti berputar
- Atur input potensiometer pada derajat tertentu
- Catat output potensiometer berhenti pada derajat apa (mendekati set derajat input potensiometer)

Tabel percobaan :

output cursor position in degrees		
required	actual	missalignment
60	40	20
70	55	15
155	135	20
210	230	20
225	115	10

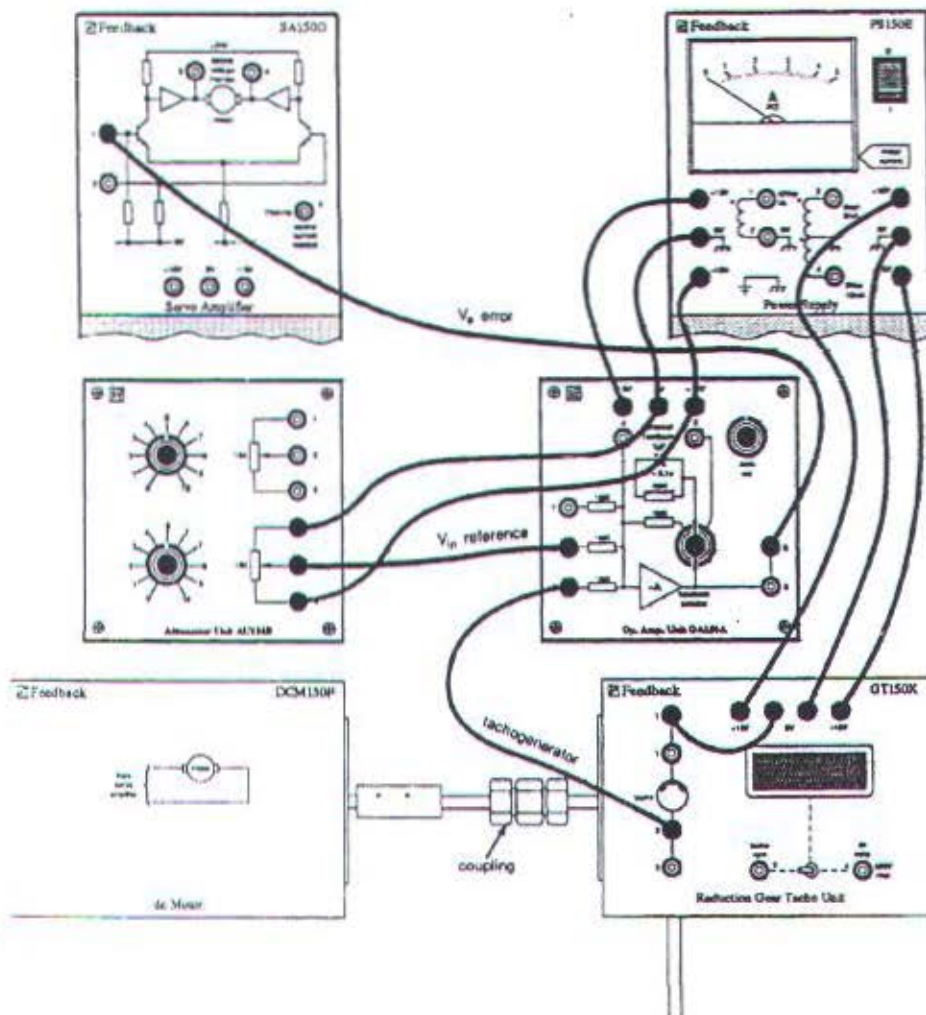
Dari tabel diatas dapat diketahui tentang missalignment antara set point dengan hasil dari sudut shaft dari input dan output potensiometer hal ini disebut dengan deadband.

5.5. Simple Speed control system

Tujuan :

- mengetahui simple control speed constructed dan operate
- mengetahui pentingnya tachogenerator dalam closed loop speed control system

gbr rangkaian :



Langkah- langkah :

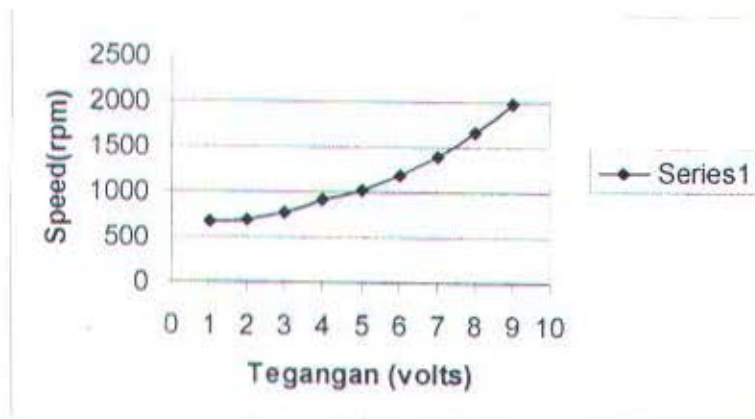
- Atur reference input nol
- Kemudian naikkan input sampai kecepatan motor sampai 2000 rpm

Tabel hasil percobaan :

Reference Voltage	Tachogenerator Voltage	Error Voltage	Speed rpm
0	1.78	3.76	660
1	1.83	3.76	690

Tabel hasil percobaan :

Reference Voltage	Tachogenerator Voltage	Error Voltage	Speed rpm
0	1.78	3.76	660
1	1.83	3.76	690
2	2.06	3.76	770
3	2.46	3.76	910
4	2.76	3.76	1030
5	3.2	3.76	1190
6	3.75	3.76	1400
7	4.45	3.76	1660
8	5.28	3.76	1970

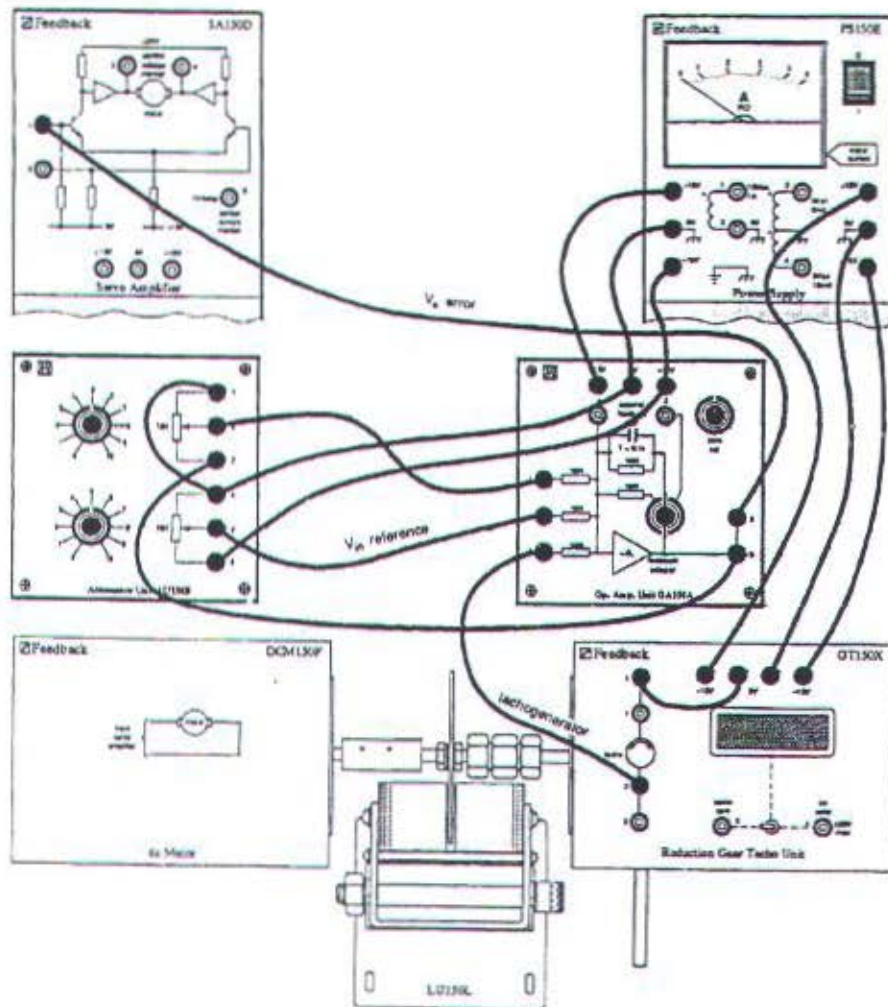


Grafik kecepatan terhadap input

Dari grafik diatas menunjukkan kemiringan dari system,kenaikan dari input proportional dengan kenaikan kecepatan putaran ,pada percobaan op amp digunakan sebagai pembanding/comparator yang hasilkan input signal ke servoamplifier untuk nilai yang diinginkan.sistem ini disebut closed loop menggunakan feedback yang akan menghasilkan error voltage.

5.5.1 Efek dari Gain pada perubahan kecepatan dan perubahan beban

Gbr rangkaian :

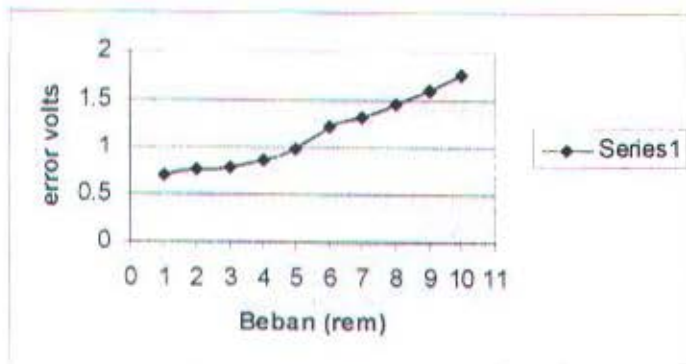


Langkah – langkah :

- atur feedback switchselector pada external feedback
- atur gain pada posisi 10
- atur rpm motor sampai 1000 rpm
- lakukan percobaan dengan berbagai macam beban mulai dari 0-10

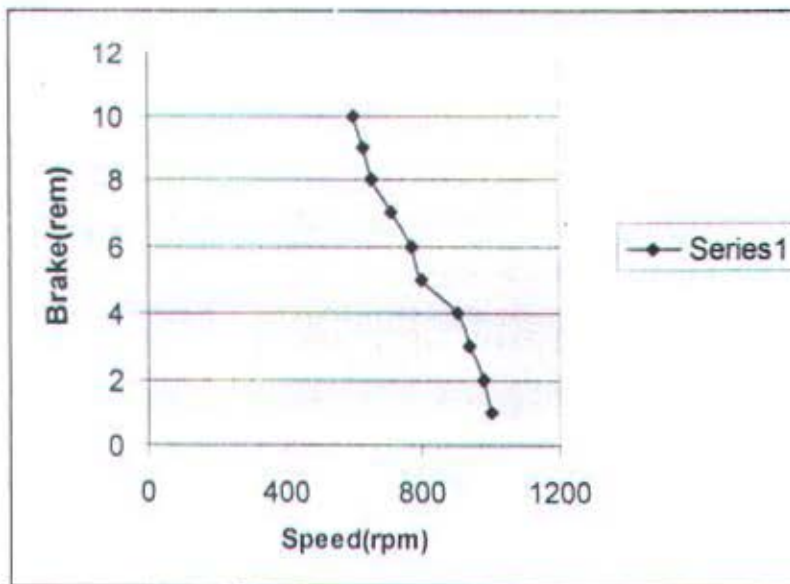
Tabel hasil percobaan :

Brake Position	Reference volts	error volts	Tachogenerator Volts	Speed rpm
1	1.79	0.7	2.69	1000
2	1.79	0.76	2.63	980
3	1.79	0.78	2.52	940
4	1.79	0.86	2.41	900
5	1.79	0.98	2.15	800
6	1.79	1.23	2.06	770
7	1.79	1.33	1.9	710
8	1.79	1.47	1.85	650
9	1.79	1.61	1.77	630
10	1.79	1.77	1.6	600



Grafik.error volts terhadap pengereman

Dari grafik diatas dapat ditarik kesimpulan bahwa pada saat gain 10 atau maximum semakin besar load akan menghasilkan arus error yang semakin besar pula.



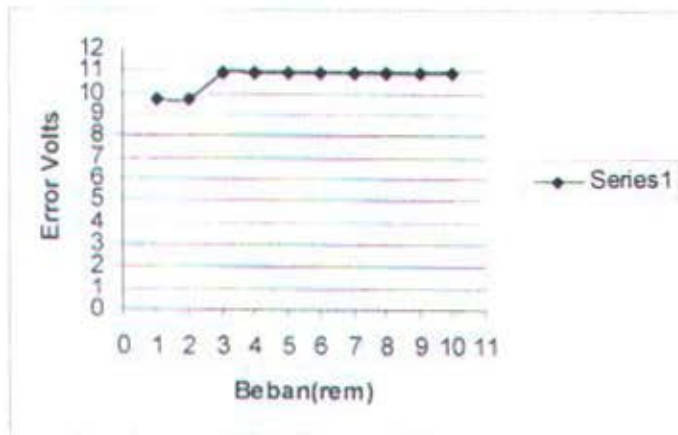
Grafik.Pengereman terhadap kecepatan

Dari grafik diatas dapat dilihat ketidaklinieran dari hasil pengamatan ,hal ini diakibatkan karena gain diset pada kondisi maximum jadi sistem tidak stabil,kesalahan baca pada alat ukur,dan juga karena set rpm pada 1000 tidak tercapai dengan sempurna.

Ulangi untuk gain = 5 yaitu posisi potensiometer pada 1

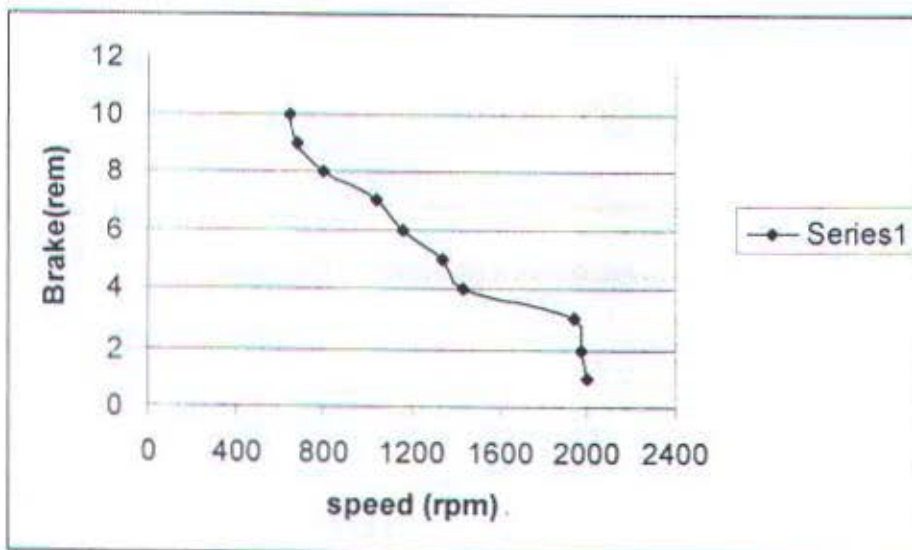
Tabel hasil percobaan :

Brake Position	Reference volts	error volts	Tachogenerator Volts	Speed rpm
1	3.85	9.67	5.3	2000
2	3.85	9.73	5.15	1970
3	3.85	10.97	3.83	1940
4	3.85	10.97	3.6	1430
5	3.85	10.97	3.1	1340
6	3.85	10.97	2.78	1160
7	3.85	10.97	2.3	1040
8	3.85	10.97	1.82	800
9	3.85	10.97	1.71	680
10	3.85	10.97	1.6	640



Grafik. Error volts terhadap pengereman

Dari grafik dapat diketahui bahwa pada saat set gain 5 error yang ada stabil. Meskipun ada perubahan load.



Grafik. Pengereman terhadap kecepatan

Dari grafik diatas dapat disimpulkan ketidak linieran disebabkan karena kesalahan baca pada alat ukur., setting rem yang tidak pas.

Langkah- langkah :

- atur rem pada posisi nol
- atur gain pada posisi 10
- pelan – pelan naikan input reference value sampai motor mulai jalan dan ukur input referensi

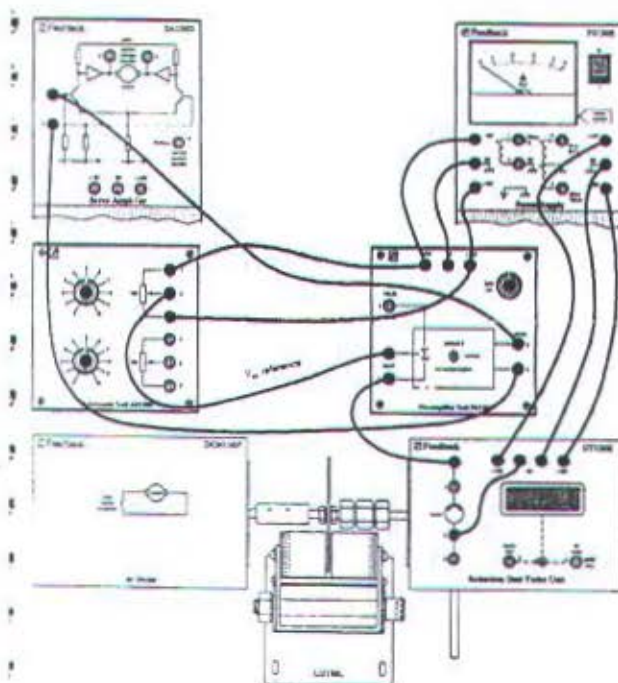
tabel hasil percobaan :

Brake	Gain	Ref input
0	10	0.3
1	1	0.1
10	5	0.25

Dari tabel diatas kita dapat menarik kesimpulan bahwa deadband berkurang apabila nilai gain naik

5.5.2 Simple reversible Speed control system

Gbr rangkaian :



Langkah – langkah :

- atur pre amp pada ac compensation, untuk kurangi riak pada pembacaan tachogenerator yang bisa menyebabkan ketidak stabilan.
- Atur potensiometer AU 150 B pada posisi 5
- Tanpa rem dimotor lakukan percobaan untuk bisa mengatur arah putaran dari motor

Tabel hasil percobaan :

Minimum signal needs for motor response	
Forward	Reverse
14.51	14.77

Dari tabel diatas maka kita dapat mengetahui minimum signal yang ada agar dapat memutar arah motor baik yang kearah depan maupun yang kearah sebaliknya.

Untuk mengetahui error voltage:

atur kecepatan pada satu arah kurang lebih 1000 rpm, variasikan dengan berbagai macam beban. kemudian balik arahnya

Tabel hasil percobaan :

Brake Position	Tachogenerator Volts	Reference Volts	Error Volts	Speed rpm	Tachogenerator Volts	Reference Volts	Error Volts	Speed rpm
0	2.67	14.69	0.95	1000	2.87	14.54	1.61	1000
1	2.39	14.68	0.95	890	2.61	14.54	1.61	980
2	2.15	14.68	0.95	810	2.32	14.54	1.61	860
3	1.83	14.68	0.95	680	2	14.54	1.61	750
4	1.61	14.68	0.95	600	1.77	14.54	1.61	360
5	1.5	14.68	0.95	560	1.57	14.54	1.61	590
6	1.39	14.68	0.95	520	1.47	14.54	1.61	550
7	1.31	14.68	0.95	490	1.35	14.54	1.61	500
8	1.15	14.68	0.95	430	1.22	14.54	1.61	450

9	1.05	14.68	0.95	390	1.11	14.54	1.81	410
10	1.01	14.68	0.95	380	1.04	14.54	1.81	390

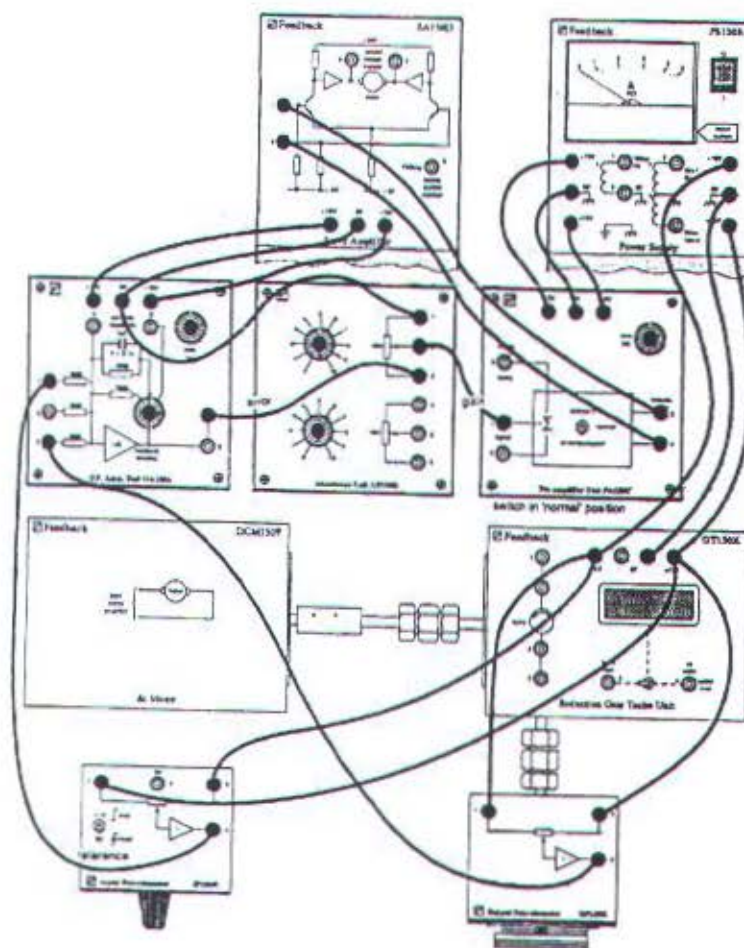
Dari tabel diatas maka kita akan dapat mengetahui besarnya arus error pada saat motor berputar searah jarum jam maupun berputar berlawanan arah jarum jam. Dimana arus errornya bertambah ketika motor di putar berlawanan arah jarum jam

5.6. Dead band

Tujuan :

- Mengetahui pengertian deadband

Gbr rangkaian :

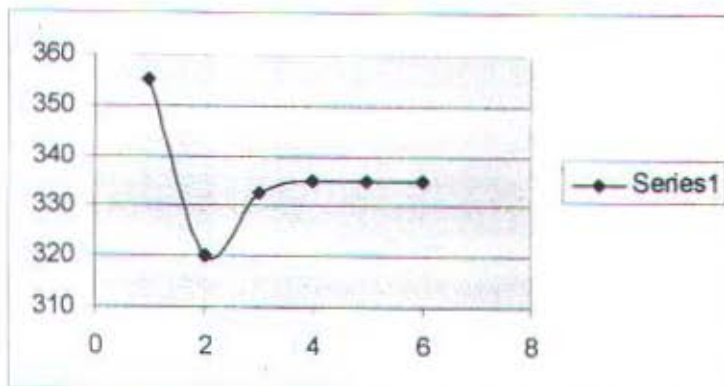


Langkah – langkah :

- atur feedback selector pada posisi $100k\Omega$
- nyalakan power supply
- atur input dan output potensiometer pada posisi nol derajat
- atur AU 150 B pada posisi 0
- atur PA150 C pada zero set supaya motor tidak berputar
- atur gain potensio pada 1
- pelan-pelan putar input potensiometer searah jarum jam sampai terjadi respon pada output potensiometer, kemudian putar berlawanan jam
- ulangi dengan bervariasi gain
- setiap percobaan reset input dan output potensiometer pada nol

Tabel hasil percobaan :

Gain	Rotation in degrees		Total Deadband degrees
	clockwise	anti clockwise	
1	200	155	355
2	165	155	320
3	162.5	170	332.5
4	180	155	335
5	180	155	335
6	180	155	335

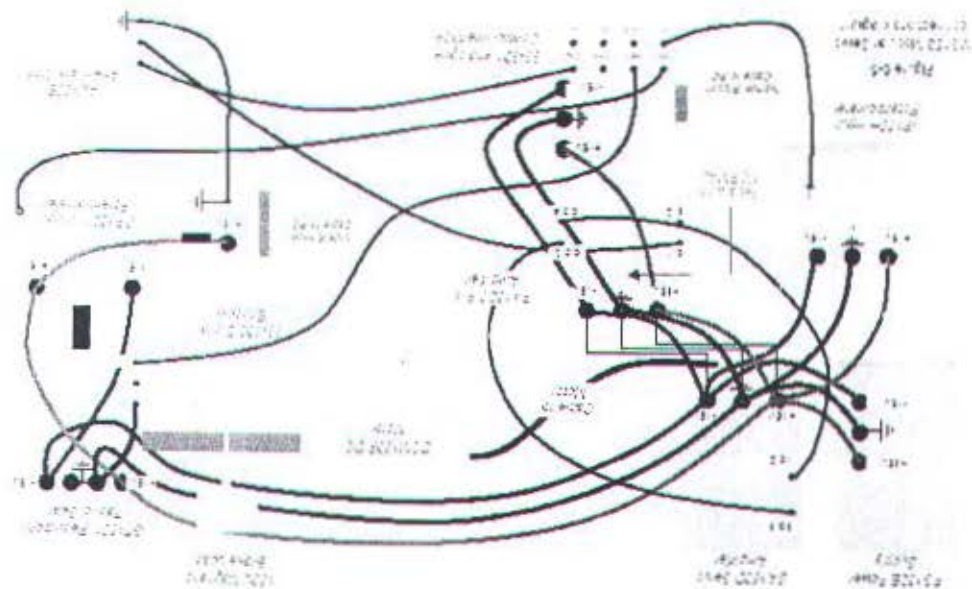


Grafik. total deadband terhadap gain

Dari grafik seharusnya deadband menurun ketika gain meningkat, hal ini mungkin karena salah pembacaan atau ada error dari voltmeter.

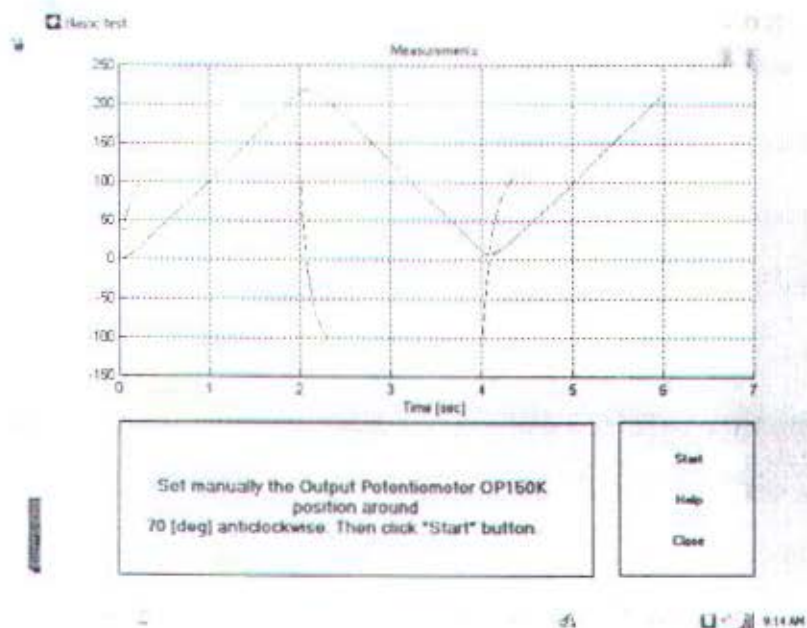
*Percobaan Berbasis Komputer :

Gbr Rangkaian :



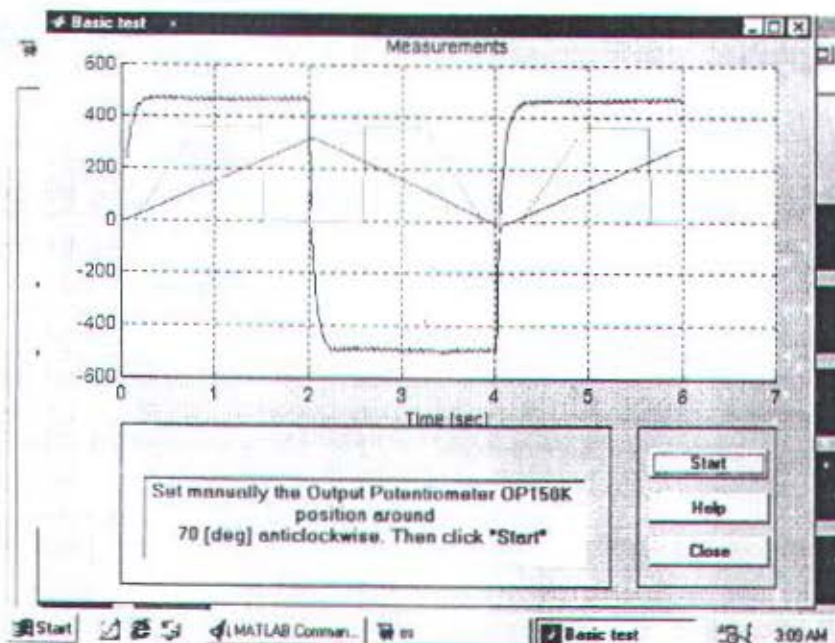
5.7 Basic test

Dalam hal ini dilakukan percobaan dengan menggunakan computer terhadap MSW 150.



Percobaan menggunakan Base address 0 yang merupakan hasil penyimulasian percobaan dengan MSW 150.

Dari grafik diatas menunjukkan kelinearitas dari performance dari system.



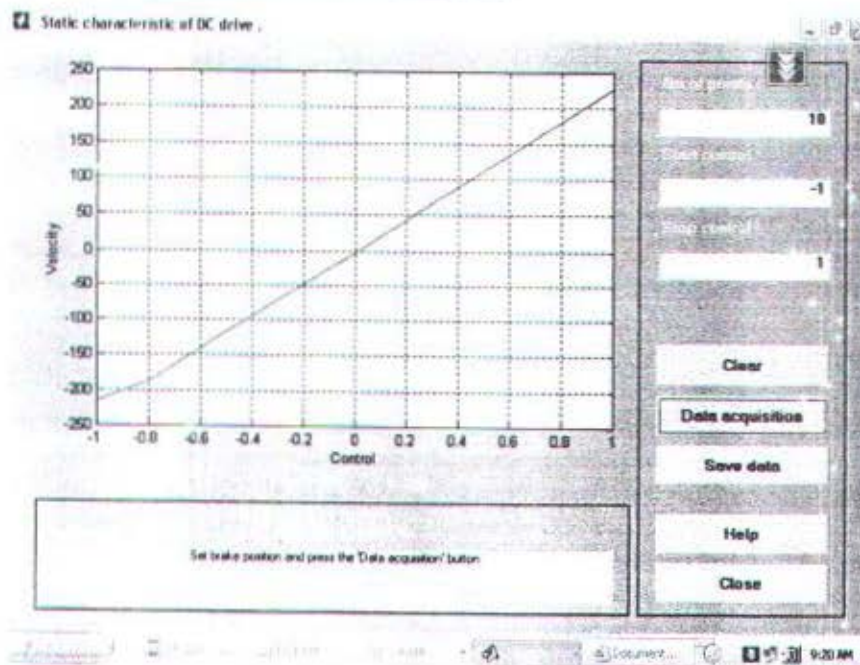
Percobaan menggunakan Base address 544 yang merupakan hasil penyimulasian percobaan MSW 150.

Dari grafik diatas menunjukkan kenonlinearitas dari performance dari system MSW 150.

Dimana :

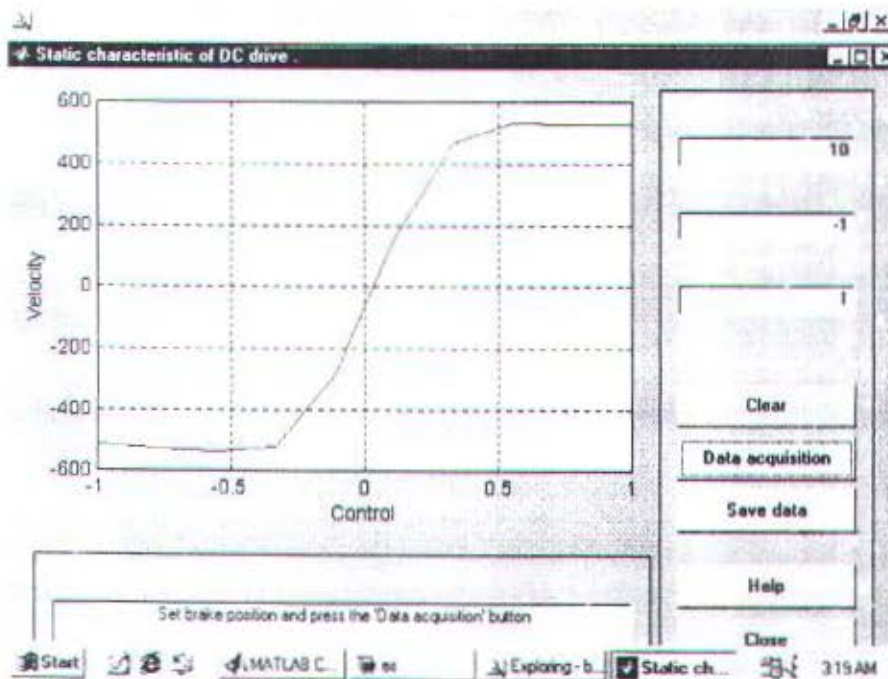
- Garis merah = channel 1 ... merupakan sudut yang diukur dari A/D converter (Gunakan A/D1)channel analog control device
- Garis Biru = channel 2....merupakan sudut yang diukur dari digital encoder
- Garis hijau = channel 3...merupakan velocity yang diukur dari A/D 2 dari analog control interface.
- Garis hitam = channel 4 ...Velocity yang diukur encoder measurement.

5.8 Static Characteristik dari Motor DC



Percobaan menggunakan Base address 0 yang merupakan hasil penyimulasian percobaan MSW 150.

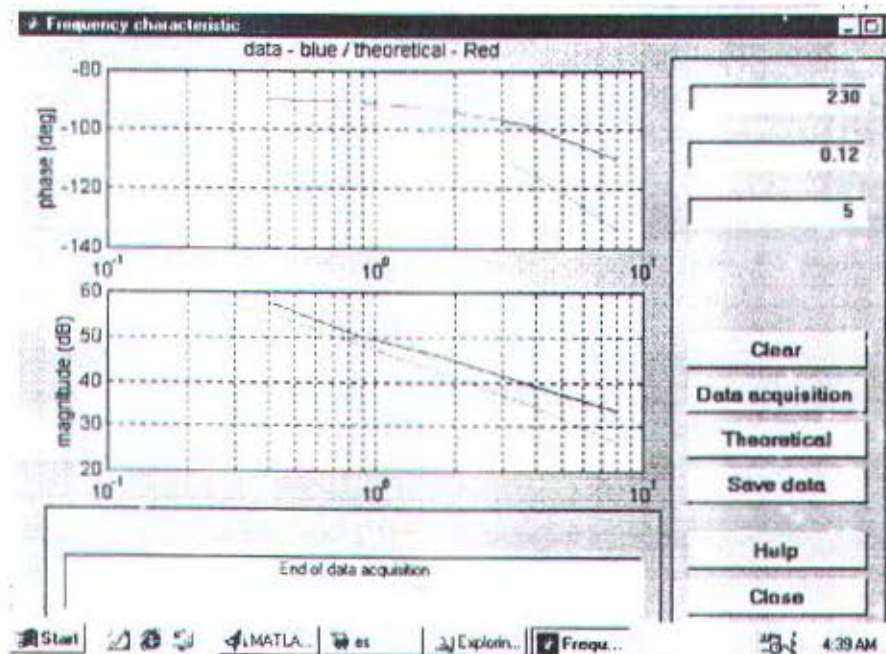
Dari grafik diatas menunjukkan kelinearitas dari performance dari Motor DC.



Percobaan menggunakan Base address 544 yang merupakan hasil penyimulasian percobaan MSW 150.

Dari grafik diatas menunjukkan kenonlinearitas dari performance dari System MSW 150.

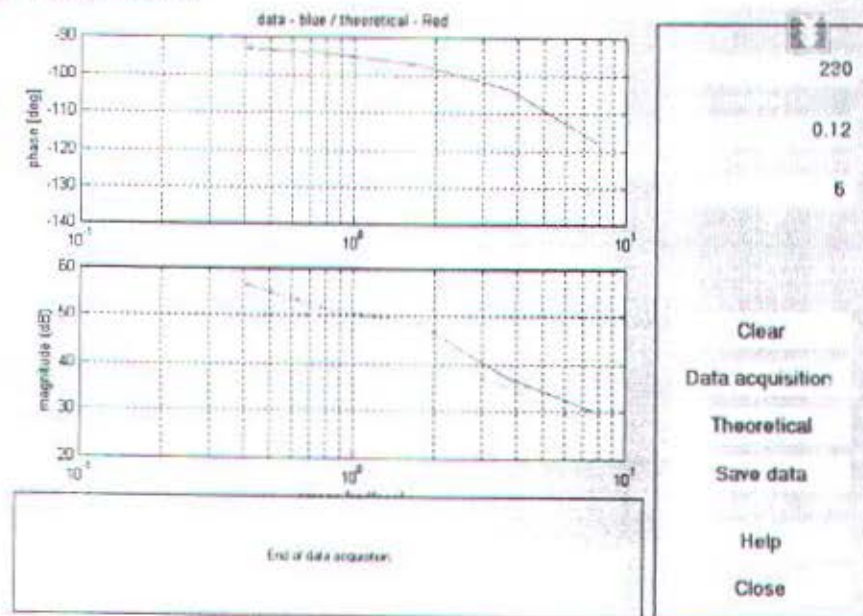
5.9 Frequency Characteristik dari Motor DC



Ini adalah percobaan menggunakan Base address 0 yang merupakan hasil penyimulasian percobaan MSW 150.

Dari grafik diatas menunjukkan kelinearitas dari performance dari Motor DC.

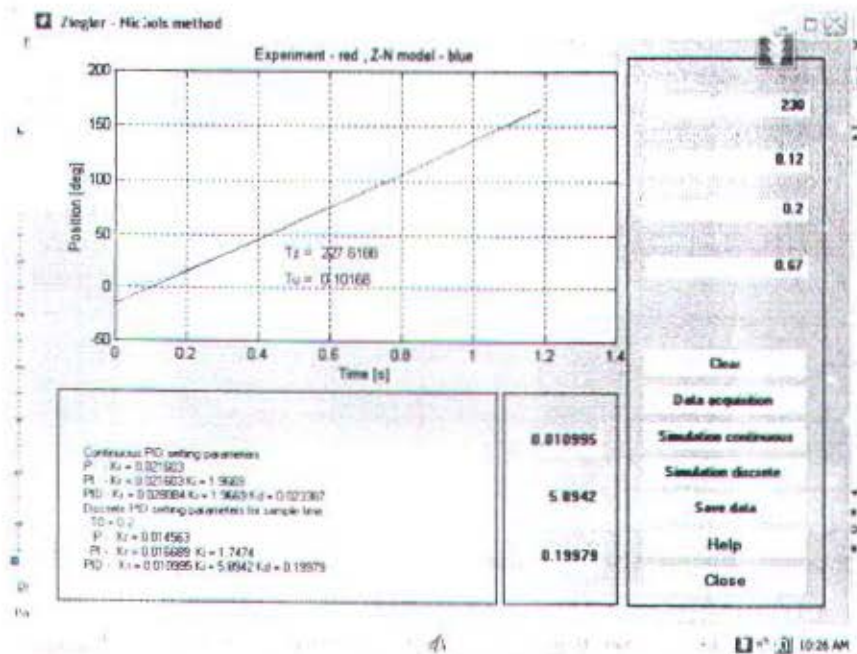
Frequency characteristic



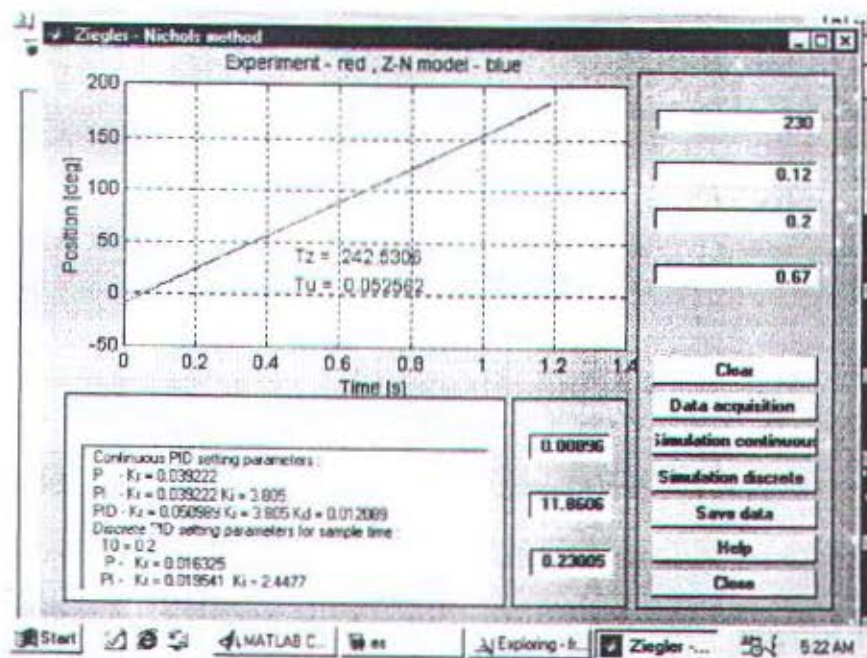
Ini adalah percobaan menggunakan Base address 544 yang merupakan hasil penyimulasian percobaan MSW 150.

Dari grafik diatas menunjukkan kenonlinearitas dari performance dari System MSW 150.

5.10 Ziegler & Nichols Method



Ini adalah percobaan menggunakan Base address 0 yang merupakan hasil penyimulasian percobaan MSW 150.

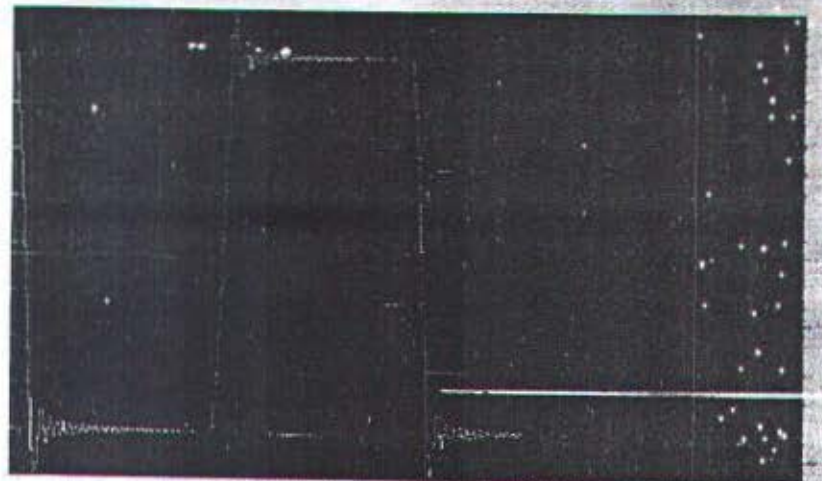


Ini adalah percobaan menggunakan Base address 544 yang merupakan hasil penyimulasian percobaan MSW 150

5.11 PID Controler

P (Proportional Controler)

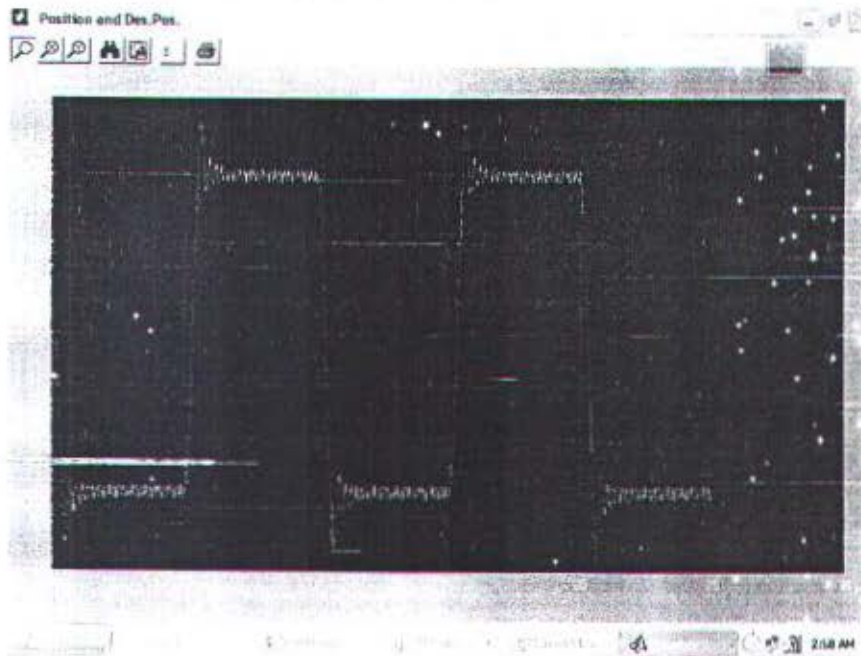
Position and Des.Pos.



Dari grafik percobaan diatas diketahui bahwa

- Waktu Turun : 4,5 detik, grafik cenderung berosilasi

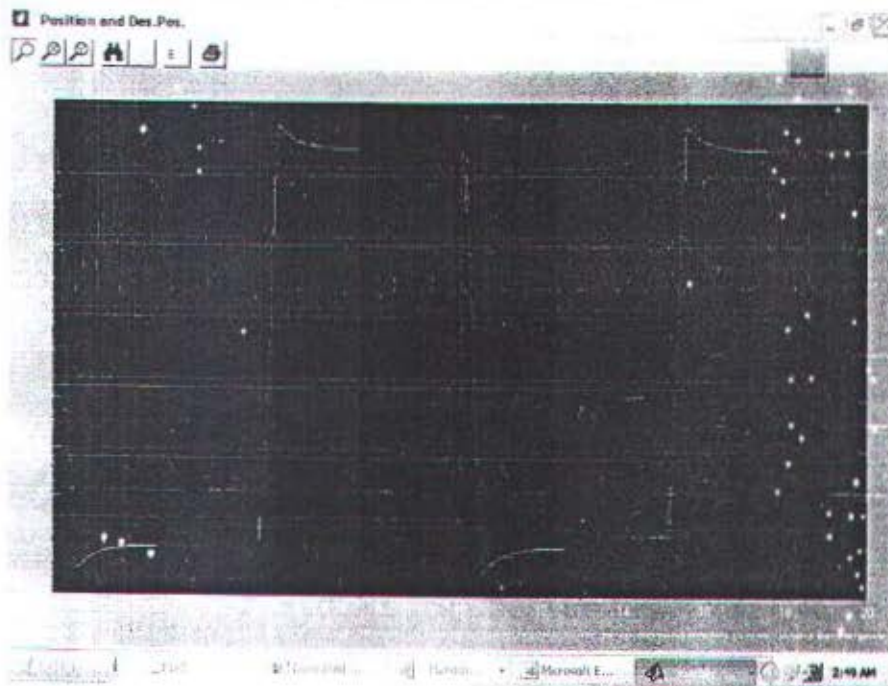
P I (Proportional + Integratif Controller)



Dari grafik percobaan diatas diketahui bahwa

- Waktu Turun : 1.5 detik, Grafik cenderung lebih berosilasi dibandingkan dengan kontrol proportional

P I D (Proportional + Integratif + Derivatif Controller)



Dari grafik Percobaan diatas diketahui bahwa:

Waktu Turun \approx 2,5 detik, grafik relative stabil meskipun kadang – kadang berosilasi

BAB VI

KESIMPULAN DAN SARAN

Dari hasil percobaan untuk komponen - komponen penunjang MSW 150 ditarik kesimpulan bahwa komponen pendukung seperti op-amp, pre-amp , attenuator, input potentiometer dan output potentiometer pada dasarnya berfungsi secara normal. Apabila sebelumnya ada kesalahan baca biasanya akibat human error/ salah pembacaan maupun pengaturan.

Aksi kontrol yang paling memberikan kestabilan pada system MS 150 adalah penggunaan controller PID (Proportional plus Integrative plus Derivative), dimana dapat dilihat pada grafik hasil percobaan bahwa waktu yang diperlukan untuk mencapai stabil lebih cepat dari sistem pengontrolan yang lain dan juga dari grafik dapat dilihat kalau pengontrolan PID relative jarang mengalami fluktuasi atau yang disebut osilasi dibandingkan dengan sistem pengontrolan yang lain.

Saran

Dalam melakukan percobaan agar secara teliti dan hati – hati dalam merangkai rangkaian .Kalibrasikan selalu alat-alat ukur agar hasil yang dicapai benar dan akurat. Perlu dibuktikan apakah aksi kontrol pada modular servo ini dapat menghasilkan hasil yang sama apabila menggunakan motor ac.

DAFTAR PUSTAKA

1. Ogata, Katsuhiko; "Teknik Kontrol Automatik (Sistem Pengaturan) I diterjemahkan oleh Edi Laksono"; Erlangga; Jakarta; 1985
2. Gunterus, Frans ; "Falsafah Dasar : Sistem Pengendalian Proses"; Elex Media komputindo; Jakarta; 1994.
3. FEED BACK Instrumen ltd, "MODULAR SERVO SYSTEM 150:England 2000.
4. Johnson,Curtis,*Process Cotrol Instrumentation Technology*,Prentice hall International Edition ,INC,Englewood Cliffs,New Jersey ,1982
5. Holman,JP, *Experimental Methods for Engineers*,McGraw-Hill ,New York, 1972.
6. Cooper,W.D and Helfrick ,D.A ,*Electoric Instrumentation and measurement Techniques*,Prentice- hall of india ,New Delhi,1985

Function untuk memanggil main control window dari MSW 150

```
function Es_model()
```

```
% This is the machine-generated representation of a Handle Graphics object  
% and its children. Note that handle values may change when these objects  
% are re-created. This may cause problems with any callbacks written to  
% depend on the value of the handle at the time the object was saved.  
%
```

```
% To reopen this object, just type the name of the M-file at the MATLAB  
% prompt. The M-file and its associated MAT-file must be on your path.
```

```
load Es_model
```

```
a = figure('Units','normalized', ...  
    'Color',[0.45 0.66 0.143], ...  
    'Colormap',mat0, ...  
    'CreateFcn','IfOpWin("Set build-in model parameters");', ...  
    'MenuBar','none', ...  
    'Name','Set build-in model parameters', ...  
    'NumberTitle','off', ...  
    'PointerShapeCData',mat1, ...  
    'Position',[0.21875 0.345 0.55125 0.343333], ...  
    'Tag','Fig1');
```

```
b = uicontrol('Parent',a, ...  
    'Units','normalized', ...  
    'BackgroundColor',[0.5 0.5 0.5], ...  
    'Position',[0.127272 0.20155 0.445455 0.654971], ...  
    'Style','frame', ...  
    'Tag','Frame1');
```

```
b = uicontrol('Parent',a, ...  
    'Units','normalized', ...  
    'BackgroundColor',[0.5 0.5 0.5], ...  
    'FontSize',10, ...  
    'FontWeight','demi', ...  
    'ForegroundColor',[1 1 1], ...  
    'Position',[0.179545 0.58 0.136364 0.111111], ...  
    'String','Ks :', ...  
    'Style','text', ...  
    'Tag','StaticText1');
```

```
b = uicontrol('Parent',a, ...  
    'Units','normalized', ...  
    'BackgroundColor',[0.5 0.5 0.5], ...  
    'FontSize',10, ...  
    'FontWeight','demi', ...  
    'ForegroundColor',[1 1 1], ...  
    'Position',[0.179545 0.32 0.136364 0.116959], ...  
    'String','I's :', ...  
    'Style','text', ...  
    'Tag','StaticText2');
```

```
b = uicontrol('Parent',a, ...
```



```

'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'CreateFcn','rem = es_call("getmodelp");rem = num2str(rem(1));set(gcbo,
"String",rem);', ...
'FontSize',10, ...
'FontWeight','demi', ...
'HorizontalAlignment','right', ...
'Position',[0.352273 0.6 0.181818 0.116959], ...
'String','230', ...
'Style','edit', ...
'Tag','EditKs');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'CreateFcn','rem = es_call("getmodelp");rem = num2str(rem(2));set(gcbo,
"String",rem);', ...
'FontSize',10, ...
'FontWeight','demi', ...
'HorizontalAlignment','right', ...
'Position',[0.352273 0.35 0.181818 0.116959], ...
'String','0.11', ...
'Style','edit', ...
'Tag','EditTs');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'BackgroundColor',[0.5 0.5 0.5], ...
'Position',[0.629545 0.20155 0.247727 0.651163], ...
'Style','frame', ...
'Tag','Frame2');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback','mat2', ...
'FontSize',10, ...
'FontWeight','demi', ...
'Position',[0.656818 0.668605 0.195455 0.127907], ...
'String','SET', ...
'Tag','Pushbutton1');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback','helpwin es_mdl_hlp', ...
'FontSize',10, ...
'FontWeight','demi', ...
'Position',[0.656818 0.459302 0.195455 0.127907], ...
'String','HELP', ...
'Tag','Pushbutton2');
b = uicontrol('Parent',a, ...
'Units','normalized', ...
'Callback','close;', ...
'FontSize',10, ...
'FontWeight','demi', ...

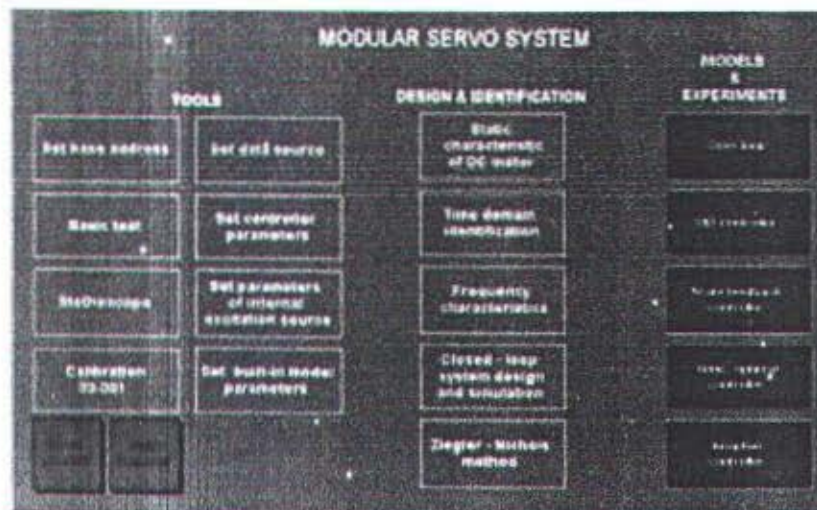
```



```

'Position',[0.656818 0.255814 0.195455 0.127907], ...
'String','CLOSE', ...
'Tag','Pushbutton3');

```



Gbr.main control window MSW 150

Function untuk percobaan PID controller

```
function [sys, x0, str, ts] = sfunc( t, x, u, flag, Par, T0, ds, exc_src, SetRTKPar, flag9 )
```

```
%
```

```
% S-function pid_sf.M - S-function for PID controller
```

```
%
```

```
% Parameters:
```

```
% t      - time,
```

```
% x      - state vector,
```

```
% u      - input to the S-function block,
```

```
% flag   - the value passed to S-function by SIMULINK to distinguished  
%          different actions. The arguments t, x, u and flag are set and  
%          passed to S-function by SIMULINK automatically,
```

```
% downsamp - downsampling ratio. Defines how many samples is transfered to  
%          the output of the S-function block. If downsamp is equal  
%          to 10 only 1 sample of every 10 samples is transfered from  
%          Real Time Task to the output of the S-function block.
```

```
% exc_src - simulink generator flag. If this value is equal to two  
%          simulink generator sets desired position of the cart. The  
%          simulink generator is connected to the input of the S-function  
%          block.
```

```
% stop_pract - stop practical after simulation/stop
```

```
% Global variables
```

```
global history es_flag9
```

```
global local_downsamp
```

```
switch flag
```

```
case 0, % Initialization
```

```
    % Set number of continuous states, number of discrete states, number of  
    % outputs and number of inputs.
```

```
    % 0 continuous states, 1 discrete state, 7 outputs, 1 inputs
```

```
    sizes.NumContStates = 0;
```

```
    sizes.NumDiscStates = 1;
```

```
    sizes.NumOutputs    = 7;
```

```
    sizes.NumInputs     = 1;
```

```
    sizes.DirFeedthrough = 0;
```

```
    sizes.NumSampleTimes = 1;
```

```
    sys = simsizes(sizes);
```

```
    local_downsamp = ds;
```

```
    es_flag9 = flag9;
```

```
    % Set initial values of the control algorithm
```

```
    if ( eq( SetRTKPar, 1 ) )
```

```
        dummy = es_call( 'SetAlgNo', 0 );
```

```
        dummy = es_call( 'SetSampleTime', T0 );
```

```
        dummy = es_call( 'SetF', Par );
```

```
        dummy = es_call( 'SetInitCond', [0 0] );
```

```

dummy = es_call( 'SetAlgNo', 2 ); % PID
dummy = es_call( 'ResetTime', 0 );
p = es_call( 'GetPW' );
p( 1 : 2 ) = [ 0 0 ];
p = es_call( 'SetPW', p );

switch exc_src,
case 1, % Simulink signal generator
    dummy = es_call( 'SetDataSource', [ 8 9 13 4 5 6 ] );
    dummy = es_call( 'SetPW', [ 0 0 zeros( 1, 18 ) ] );
case 2, % External signal source
    dummy = es_call( 'SetDataSource', [ 8 9 12 4 5 6 ] );
case 3, % Built-in signal generator (square)
    dummy = es_call( 'SetDataSource', [ 8 9 13 4 5 6 ] );
    dummy = es_call( 'SetPW', [ 1 3 3 2 2 -50 50 -50 50 zeros( 1, 11 ) ] );
end

```

```

es_cpar( 'esc_UpdateButton' );
es_egen( 'esg_UpdateButton' );

```

```

end

```

```

    % Wait for the first sample which may be send to the output
    while ( es_call( 'GetNoOfSamples', 0 ) <= ds )
        ;
    end;

```

```

end;

```

```

history = es_call( 'GetHistory', 0 );

```

```

% initialize the initial conditions
str = []; % str is always an empty matrix
ts = [-2 0]; % initialize the array of sample times
    % variable sample time
    % Set initial conditions of the state
    x0 = 1;

```

```

% change background color of Real Time Task block after simulation start
set_param('es_pid/RealTimeTask','BackgroundColor','cyan');

```

```

case 1, % Unhandled flags
    sys = [];

```

```

case 2, % Calculate discrete state
    % Set desired value
    if eq( exc_src, 1 ) % Simulink signal generator
        dummy = es_call( 'SetPW', [ 0 u zeros( 1, 18 ) ] );
    end
    sys = x;

```

```

case 3, % Calculate outputs

```



```

[ sys, history ] = es_sfco( local_downsamp, history );

case 4, % Calculate next discrete time point
[ sys, local_downsamp, history ] = es_sfntp( local_downsamp, history );

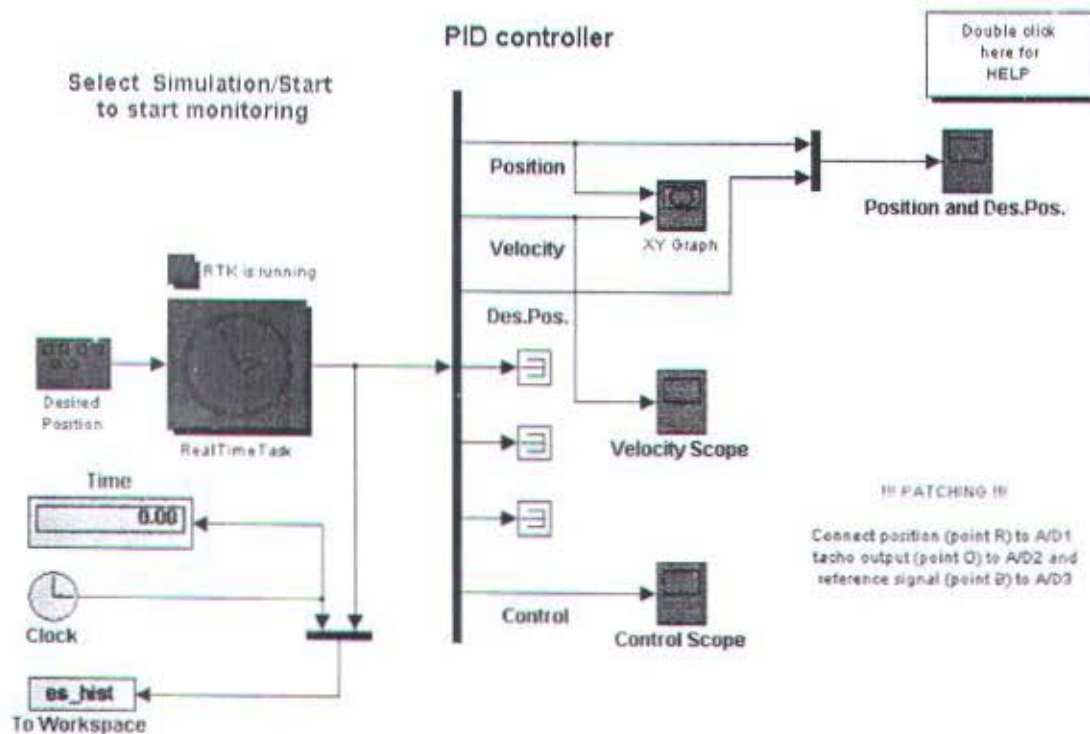
case 9, % Terminate
% change background color of Real Time Task block after simulation stop
set_param('es_pid/RealTimeTask','BackgroundColor','green');

if( eq( es_flag9, 1 ) )
    dummy = es_call( 'SetAlgNo', 0 );
    dummy = es_call( 'SetSampleTime', 0.05 );
end

otherwise % Unexpected flags %
error( [ 'Unexpected flag = ', num2str(flag) ] );

end

```



Gbr.PID Controlled window

Parameter – parameter dalam PID

Real Time Model Parameters (mask)

Parameters for the Real Time SIMULINK Model
 $U = K_r * (e + i * e / T_i + D * e * T_d)$

Parameters

Sampling time:
0.01

PID parameters (K_r, T_i, T_d):
[0.025 3.46 0.013]

Downsampling ratio:
4

Data source: Simulink

☒ Stop practical after Simulation/Stop

Apply Revert Help Close

SERVO Real-Time Task

S-Function

User-definable block. Blocks may be written in M, C or Fortran and must conform to S-function standards. t, x, u and flag are automatically passed to the S-function by Simulink. 'Extra' parameters may be specified in the 'S-function parameters' field.

Parameters

S-function name:
es_pidf

S-function parameters:
T_i par_downsamp stop_pract exc_stop

Apply Revert Help Close

Mask Editor cs_pid/RealTimeTask

Icon | Initialization | Documentation

Mask type: Real Time Model Parameters

	Prompt	Type	Variable
Add	Sampling time:	edit	T0
Delete	PID parameters (Kr, Ti, Td	edit	par
Up	Downsampling ratio:	edit	dow
Down	Data source:	popup	src
	Stop practical after simul	checkbox	sto
	<<end of parameter list>>		

Prompt: Sampling time Control type: Edit

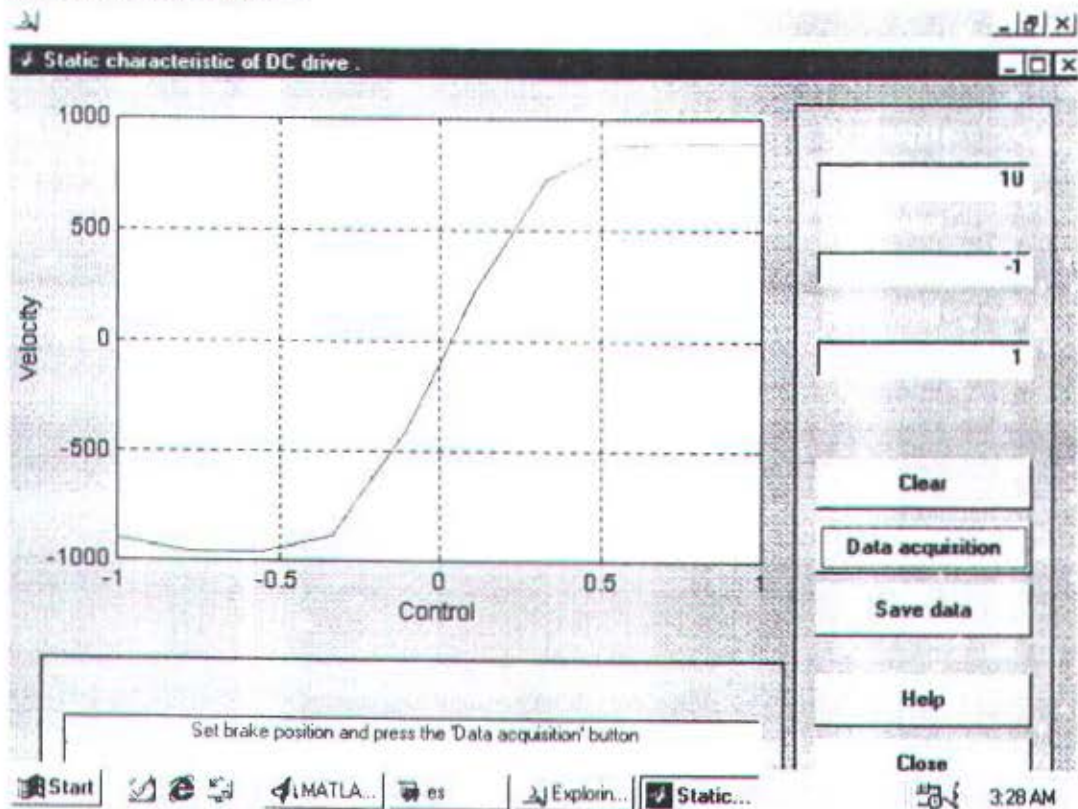
Variable: T0 Assignment: Evaluate

Popup strings:

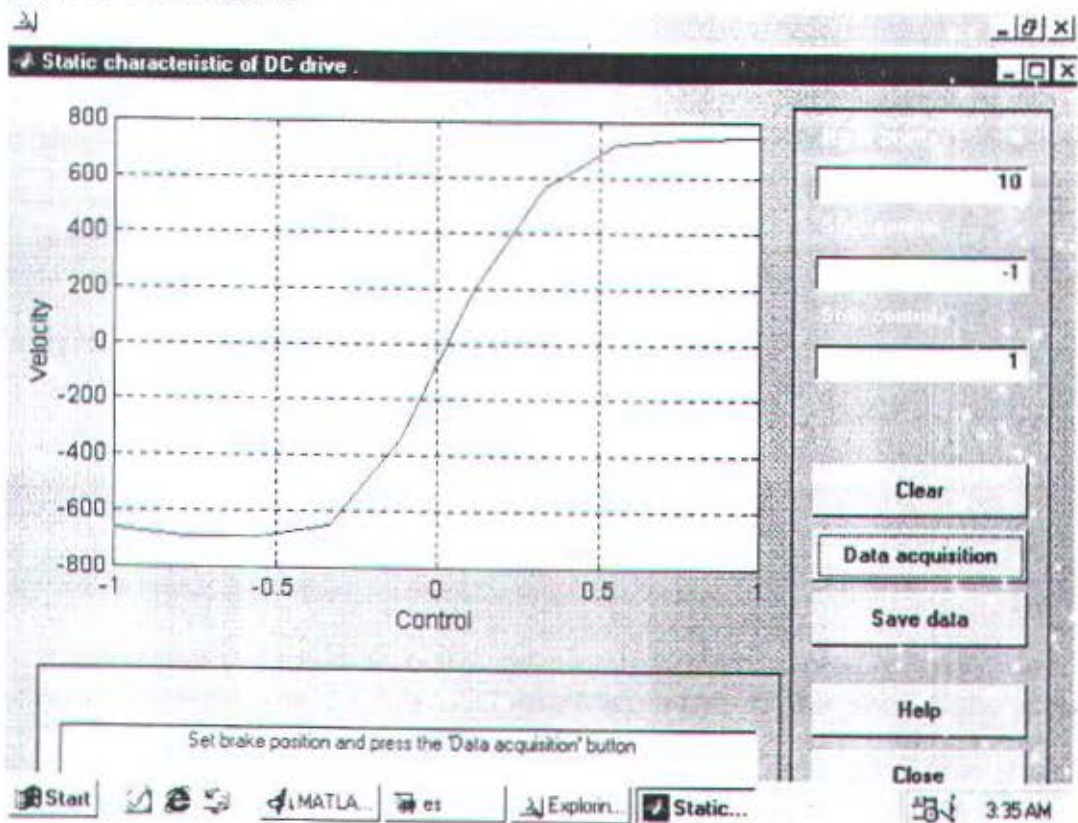
Initialization commands:

Apply | Revert | Unmask | Help | Close

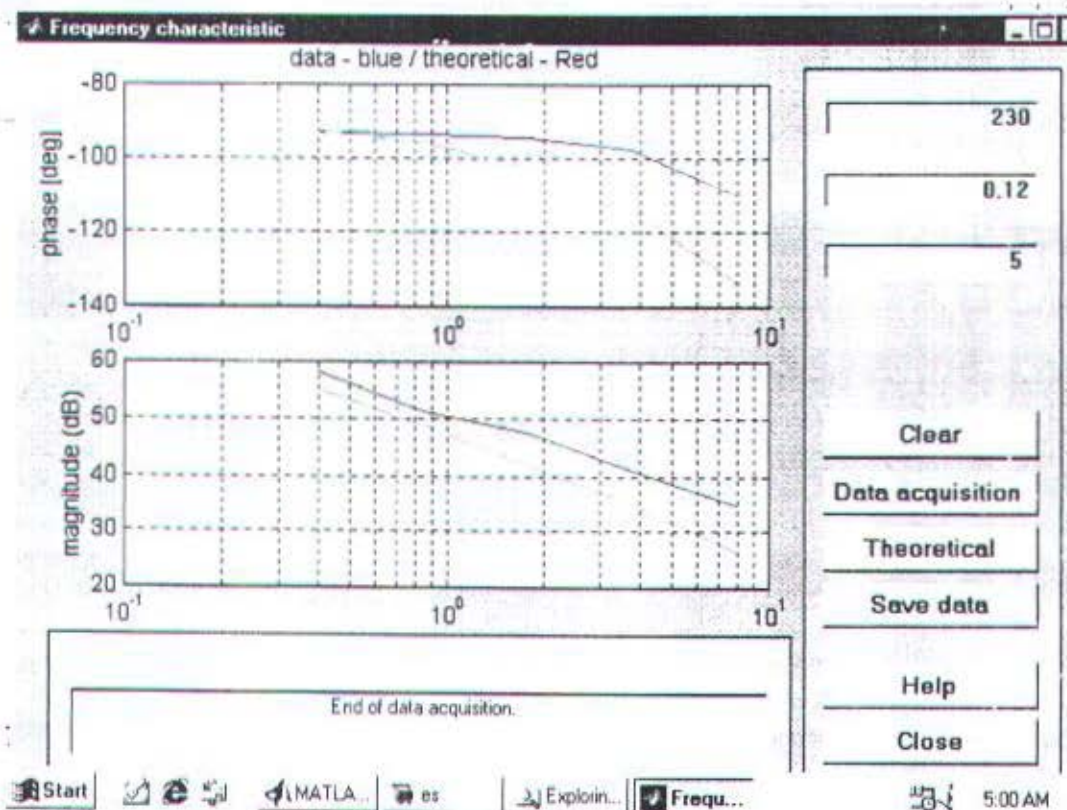
STATIC PADA REM 5



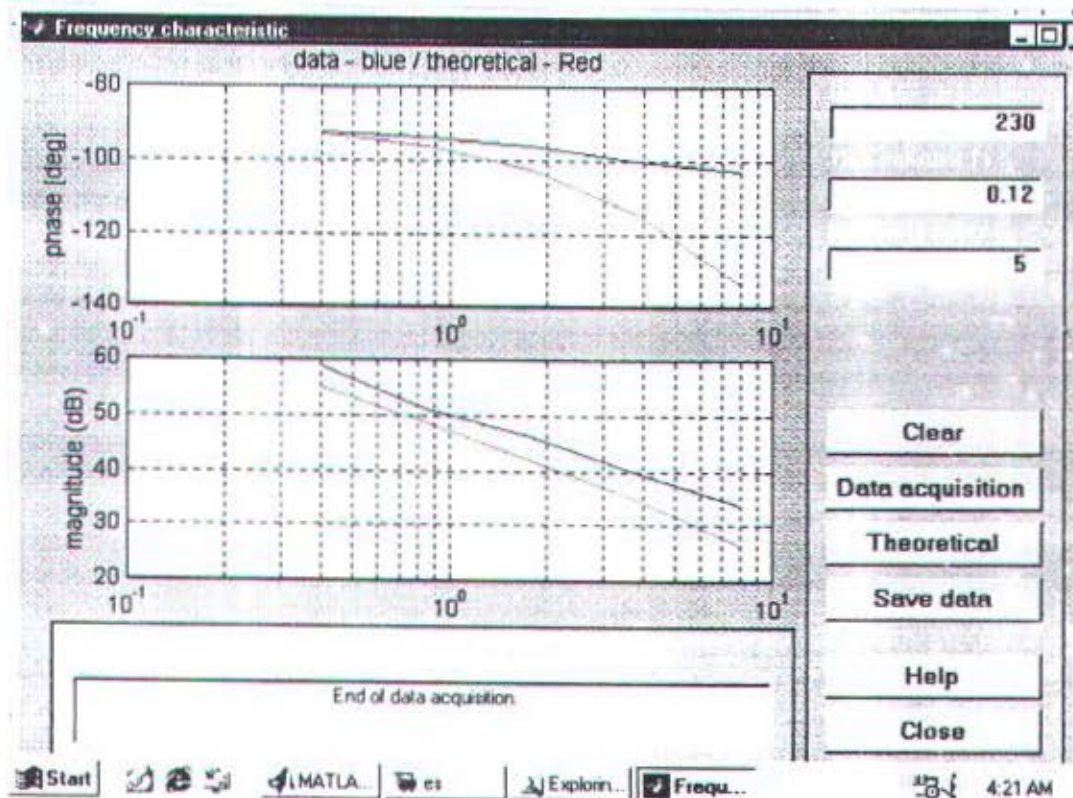
STATIC PADA REM 8



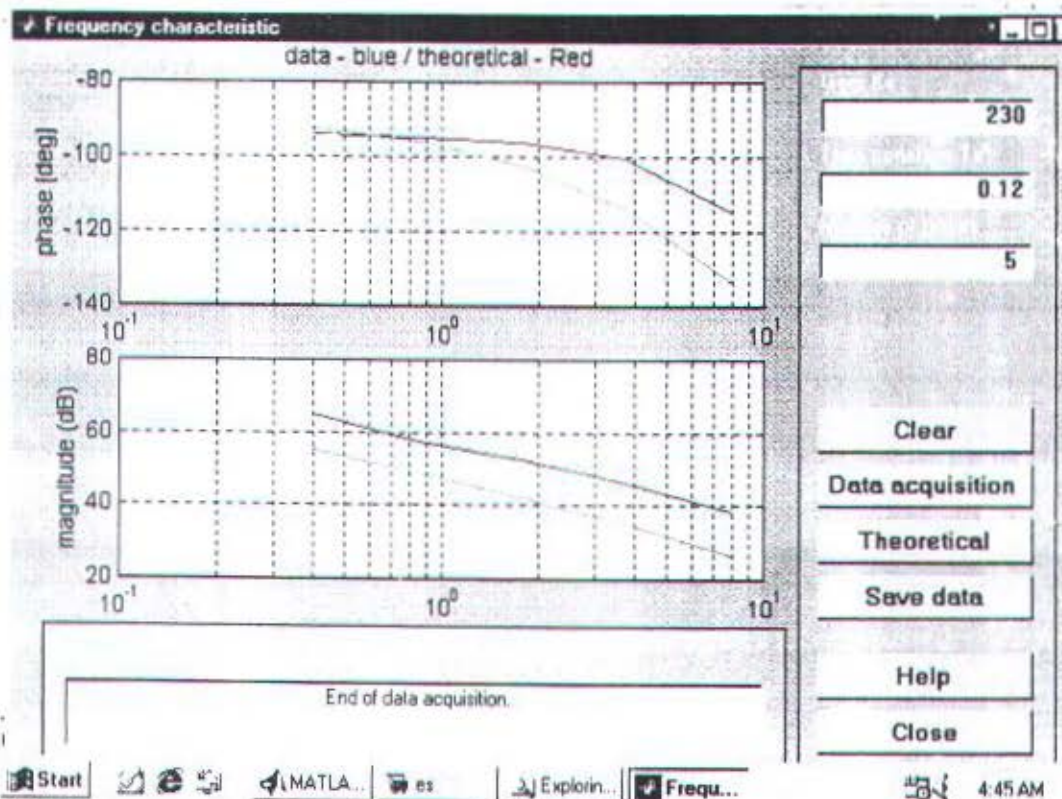
REM 5



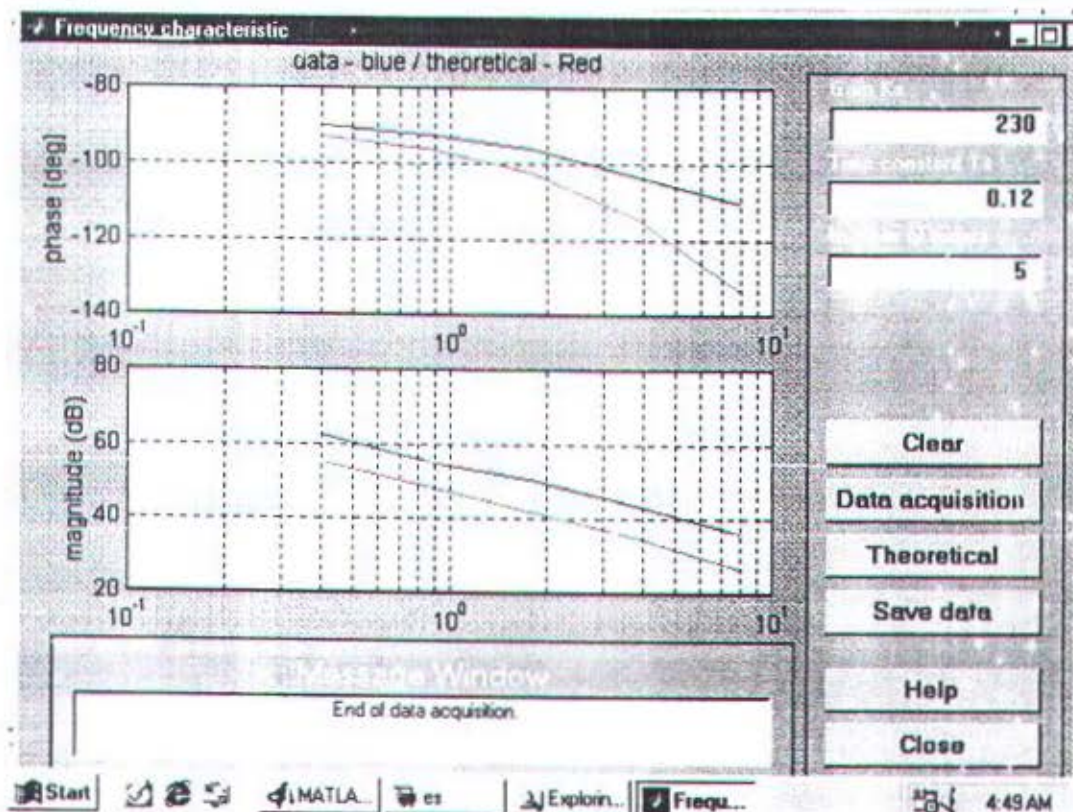
REM 8



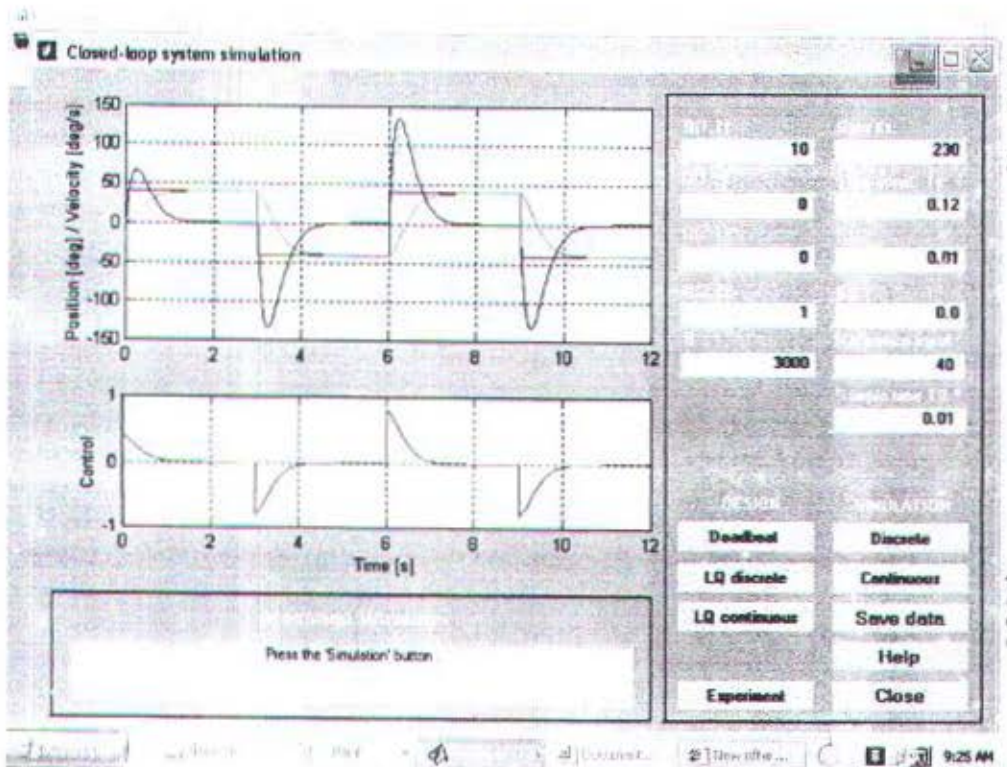
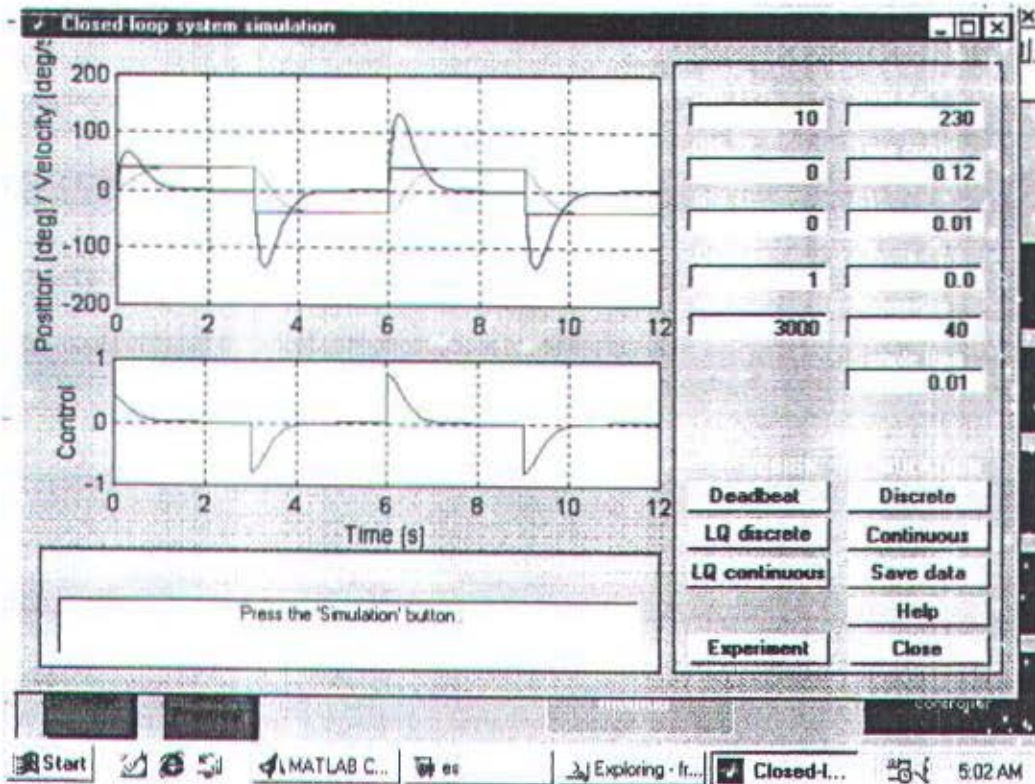
REM 1



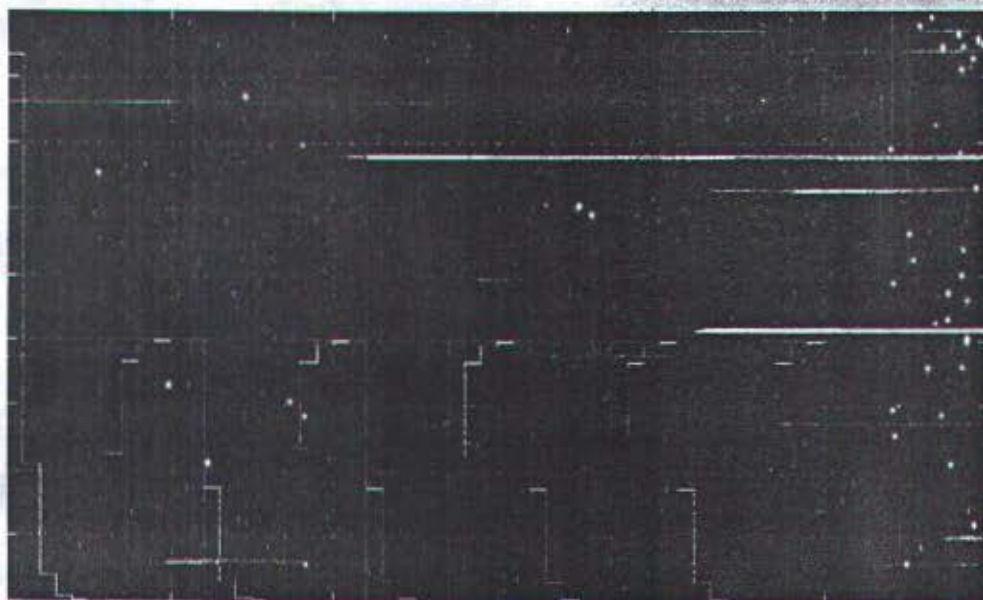
REM 2



PERCOBAAN LQ CONTROLLER



Position and Des.Pos.



9:28 AM

